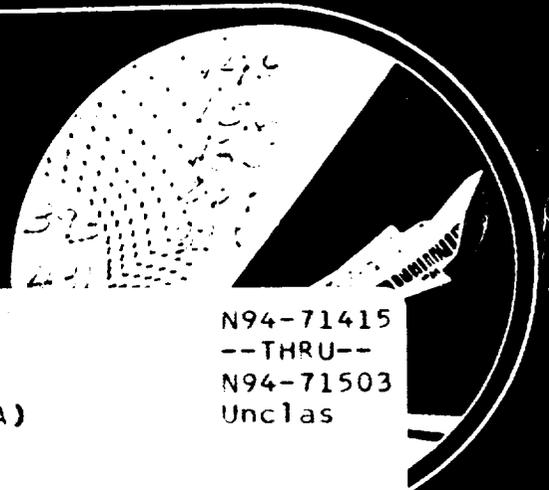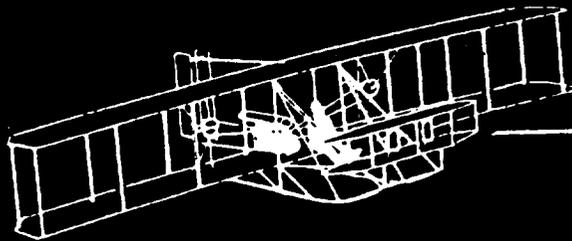NASA

THIRD AIR FORCE / NASA SYMPOSIUM ON RECENT ADVANCES IN
MULTIDISCIPLINARY ANALYSIS AND OPTIMIZATION

A COLLECTION OF TECHNICAL PAPERS

SEPTEMBER 24 - 26, 1990
SAN FRANCISCO, CA

SAN FRANCISCO AIRPORT HILTON

Third Air Force / NASA Symposium On Recent Advances in Multidisciplinary Analysis and Optimization

SAN FRANCISCO AIRPORT HILTON
24 - 26 SEPTEMBER 1990

CONFERENCE CO-CHAIRMEN

Dr. V.B. Venkayya
WRDC/FIBRA
WPAFB, OH 45433-6553
(513) 255-7191
(513) 255-6992

Dr. Jaroslaw Sobieski
NASA LaRC MS 242
Hampton, Va 23665
(804) 864-2799
(804) 864-2801

Dr. Laszlo Berke
NASA LeRC September 1990
21000 Brookpark Rd
Cleveland, OH 44135
(216) 433-3212

HOSTED BY ANAMET LABORATORIES, INC.

THIRD AIR FORCE/NASA SYMPOSIUM ON
RECENT ADVANCES IN
MULTIDISCIPLINARY ANALYSIS AND OPTIMIZATION

September 24-26, 1990/San Francisco, California

Table of Contents

SESSION 7 - Dynamics and Controls 2 (cont.)

SESSION 8 - Software Systems 1

SESSION 9 - Optimization Theory 1

vii

xvii

INTEGRATED CONTROLS-STRUCTURES DESIGN METHODOLOGY DEVELOPMENT
FOR A CLASS OF FLEXIBLE SPACECRAFT

P. G. Maghami   S. M. Joshi

J. E. Walz   E. S. Armstrong

NASA Langley Research Center
Hampton, Virginia 23665

INTRODUCTION

Future utilization of space will require large space structures in low-Earth and geostationary orbits. Example missions include: Earth observation systems, personal communication systems, space science missions, space processing facilities, etc., requiring large antennas, platforms, and solar arrays. The dimensions of such structures will range from a few meters to possibly hundreds of meters. For reducing the cost of construction, launching, and operating (e.g., energy required for reboosting and control), it will be necessary to make the structure as light as possible. However, reducing structural mass tends to increase the flexibility which would make it more difficult to control with the specified precision in attitude and shape. Therefore, there is a need to develop a methodology for designing space structures which are optimal with respect to both structural design and control design.

In the current spacecraft design practice, it is customary to first perform the structural design and then the controller design. However, the structural design and the control design problems are substantially coupled and must be considered concurrently in order to obtain a truly optimal spacecraft design. For example, let $\mathscr{C}$ denote the set of the "control" design variables (e.g., controller gains), and $\mathscr{S}$ the set of the "structural" design variables (e.g., member sizes). If a structural member thickness is changed, the dynamics would change which would then change the control law and the actuator mass. That would, in turn, change the structural model. Thus, the sets $\mathscr{C}$ and $\mathscr{S}$ depend on each other.

Future space structures can be roughly divided into four mission classes. Class I missions include flexible spacecraft with no articulated appendages which require fine attitude pointing and vibration suppression (e.g., large space antennas). Class II missions consist of flexible spacecraft with articulated multiple payloads, where the requirement is to fine-point the spacecraft and each individual payload while suppressing the elastic motion. Class III missions include rapid slewing of spacecraft without appendages, while Class IV missions include general nonlinear motion of a flexible spacecraft with articulated appendages and robot arms. Class I and II missions represent *linear* mathematical modeling and control system design problems (except for actuator and sensor nonlinearities), while Class III and IV missions represent *nonlinear* problems.

In this paper, we shall address the development of an integrated controls/structures design approach for Class I missions. The performance for these missions is usually specified in terms of (i) root mean square (RMS) pointing errors at different locations on the structure, and (ii) the rate of decay of the transient response. Both of these performance measures include the contributions of rigid as well as elastic motion.

CONTROLLER DESIGN METHODS

Control of large flexible space structures (LFSS) is a challenging problem because of their special dynamic characteristics, which include: large number of significant structural modes, low, closely-spaced structural mode frequencies, very small inherent damping, and lack of accurate knowledge of the parameters. In order to be practically

implementable, the controller must be of a reasonably low order and must also satisfy the performance specifications (i.e., RMS pointing error, closed-loop bandwidth, etc.). It must also have robustness to "nonparametric" uncertainties (i.e., unmodeled structural modes), and to "parametric" uncertainties (i.e., errors in the knowledge of the design model). The two major categories of controller design methods for LFSS are "model-based" controllers (MBC) and "dissipative" controllers.

An MBC generally consists of a state estimator (a Kalman-Bucy filter or an observer) followed by a linear-quadratic regulator (LQR). The state estimator utilizes the knowledge of the "design" model (consisting of the rigid rotational modes and a few structural modes) in its "prediction" part. Using multivariable frequency-domain design methods, such controllers can be made robust to unmodeled structural dynamics; that is, the "spillover" effect can be overcome [1]. However, such controllers generally tend to be very sensitive to uncertainties in the design model, in particular, to uncertainty in the structural mode frequencies [1,2]. An analytical explanation of this instability mechanism may be found in [2]. Achieving robustness to real parametric uncertainties is as yet an unsolved problem, although considerable research activity is in progress in that area using H-infinity and structured-singular-value methods.

In view of the sensitivity problem of MBC's, dissipative controllers, which utilize collocated and compatible actuators and sensors, offer an attractive alternative. Dissipative controllers utilize special passivity-type input/output properties of the plant, and offer robust stability in the presence of both *nonparametric* and *parametric* uncertainties. The simplest controller of this type is the *constant-gain* dissipative controller. Using collocated torque actuators and attitude and rate sensors, the constant-gain dissipative control law is given by:

$$u = -G_p y_p - G_r y_r \tag{1}$$

where y_p and y_r are the measured (3m × 1, where m is the number of 3-axis sensors) attitude and rate; G_p and G_r are 3m × 3m symmetric, positive-definite gain matrices. This control law has been proven to give guaranteed closed-loop stability despite unmodeled elastic modes, parameter errors, certain types of actuator and sensor nonlinearities, and first-order actuator dynamics [1]. The drawback of this controller is that the performance is inherently limited because of its restricted mathematical structure.

In order to obtain higher performance while still retaining the highly desirable robust stability, *dynamic* dissipative compensators can be used. Two types of such controllers were considered in [2] and are presently under development. The main characteristic of all dissipative controllers is that, although they utilize the knowledge of the design model to obtain the best possible performance, they do not rely on this knowledge to ensure stability.

INTEGRATED DESIGN FORMULATION

In order to facilitate the integrated design methodology development, an Earth-pointing geostationary platform concept was selected as a focus mission. The "Earth-pointing System (EPS)" concept, shown in Figure 1, consists of a 10-bay, 30-meter long truss structure with two radial rib antennas (7.5m and 15m diameter) at both ends. All the members including the truss, the antennas, and the antenna supports are assumed to be hollow tubes with circular cross-section and 1.59 mm thickness. The mission requirement dictates that the larger antenna be pointed to its target with a specified accuracy (i.e., the RMS error not to exceed 11 micro-radians). The antennas are assumed to be locked (i.e., fixed with respect to the truss) during normal operation, so that the problem is that of controlling the pointing and vibration of the entire structure. It is assumed that a three-axis control moment gyro (CMG) and collocated attitude and rate

2

sensors, located close to the center-of-mass of the structure, are used for accomplishing the control.

The approach followed herein is to formulate the integrated design problem as a single-objective optimization problem. The structural design variables used are outer diameters of the truss and antenna support members with the thickness fixed. In particular, the truss was broken into three sections and the outer diameters of the longerons, battens, and diagonals within each section constitute nine design variables. Two additional structural design variables are the outer diameters of the support members for the two antennas, thus constituting a total of 11 structural design variables.

The control law considered herein is the constant-gain dissipative controller given by Eq. (1), which is known to have excellent robustness to unmodeled elastic mode dynamics and parametric uncertainties. The set $\mathcal{C}$ of the control design variables consists of the controller gains G_p and G_r. In order to ensure that G_p and G_r are symmetric and positive definite, they are expressed in terms of their Cholesky decompositions:

$$G_p = \Gamma_p^T \Gamma_p \qquad\qquad G_r = \Gamma_r^T \Gamma_r$$

where Γ_p and Γ_r are upper-triangular matrices. Thus, the number of control design variables is 12, so that the total number of design variables is 23.

The sensor outputs are contaminated with zero-mean white noise processes w_p and w_r with covariance intensities W_p and W_r. It is straightforward to write the equation for the evolution of the state vector covariance matrix [3]. The steady-state version of the covariance equation is a Lyapunov equation, which can be readily solved to obtain the steady-state covariance matrix Σ of the state vector. The RMS pointing error at a given location can be determined from Σ in a straightforward manner.

The objective considered herein is to obtain the best possible performance with the least possible total mass. This is expressed as a weighted sum of the total mass and a measure of the "time constant", as:

Minimize

$$J = \beta \; \frac{(M_{struct.} + M_{act.})}{(M_{struct.}^0 + M_{act.}^0)} \; + \; (1-\beta) \; \frac{\sum_i |[1/\mathcal{R}e\{\lambda_i\}]|}{\sum_i |[1/\mathcal{R}e\{\lambda_i^0\}]|}$$

with respect to: $d_1, \ldots, d_{11}, \Gamma_p, \Gamma_r$;

where $M_{struct.}$, $M_{act.}$ denote the structural mass and the actuator mass, $\mathcal{R}e\{\lambda_i\}$ denotes the real part of the ith closed-loop eigenvalue λ_i, and the superscript "0" denotes the nominal values of the corresponding variables. The coefficient β is chosen to be between 0 and 1, according to the relative importance given to the total mass and the response "time constant", represented by the term inside the summation signs. The "time constant" term is a measure of how fast the motion (including the elastic motion) is attenuated. The reciprocal of the "time constant" term is a measure of the closed-loop performance and is called the "controlled performance".

The constraints are given by:

1) Limit on the maximum allowable RMS error at the larger antenna:

$$\varepsilon_{RMS} \leq \varepsilon_{MAX} = 11 \ \mu rad.$$

2) Limits on the minimum and maximum allowable tube diameters:

$$d_{i(MAX)} \geq d_i \geq d_{i(MIN)}, \quad i = 1, \ldots, 11$$

3) Matrices Γ_p and Γ_r must remain nonsingular (i.e., must have non-zero diagonal elements) in order to ensure positive definiteness of G_p and G_r.

INTEGRATED DESIGN RESULTS

For the nominal structural design, the first modal frequency was about 0.6 Hz at the large antenna support with the first truss mode at about 6 Hz. A 0.5 percent open-loop modal damping is assumed. The nominal control gain matrices were diagonal with elements chosen to give satisfactory closed-loop frequency and damping for the rigid-body dynamics and to maintain the RMS pointing error within the required tolerance. Optimization studies were performed using an integrated design package under development at Langley Research Center. Figures 2-5 show, respectively, the behavior (normalized relative to the nominal design) of the objective function (J), controlled performance, structural mass, and actuator mass as functions of the trade-study parameter β. A value of β near zero corresponds to a "performance" or "control" dominated design while β near 1 corresponds to a "mass" or "cost" dominated design. As β ranges between 0 and 1, Figures 2-5 indicate the trade-offs between structural and control properties of the optimal integrated design. For $\beta = 0.15$, Table I compares the corresponding integrated design with a control-optimized (i.e., conventional) design in which the structural parameters were held fixed at their nominal values and the controller parameters are chosen to minimize J. The data in Table I shows that the integrated design approach produces a lighter, more flexible structure with greatly improved performance. Not only does the integrated design approach reduce the structural mass, but the mass is also redistributed. Structural mass is removed from the truss section and added to the antenna supports. The large antenna support modal frequency was increased to near 2 Hz with a reduction of truss-mode frequencies by as much as 50 percent at the higher end of the spectrum. However, the main advantage of integrated design is in its capability to obtain a better design and not necessarily reduction of the total mass.

CONCLUDING REMARKS

The integrated controls/structures design problem was formulated as a single-objective constrained optimization problem with the structural member sizes and the control gain as the design variables. Based on the numerical results obtained for a geostationary platform model, the integrated design approach gave significantly superior designs compared to the conventional control-optimized design. Our future efforts are being directed towards developing more advanced control laws (e.g., dissipative dynamic compensators) for incorporation in the integrated design method and towards experimental verification of the method. Methods for optimal placement of sensors and actuators are also under development [4].

REFERENCES

1. Joshi, S. M.: *Control of Large Flexible Space Structures*. Berlin Springer-Verlag, 1989 (Vol. 131, Lecture Notes in Control and Information Sciences).

2. Joshi, S. M., and Maghami, P. G.: Dissipative Compensators for Flexible Spacecraft Control. Proc. 1990 American Control Conference, San Diego, CA, May 23-25, 1990.

3. Stengel, R. F., *Stochastic Optimal Control: Theory and Applications*, John Wiley & Sons, Inc., New York City, 1986.

4. Maghami, P. G., and Joshi, S. M.: Sensor/Actuator Placement for Flexible Space Structures. Proc. 1990 American Control Conference, San Diego, CA, May 23-25, 1990.

Table 1. Conventional Design vs. Integrated Design

	Objective Function	Controlled Performance	Structural Mass	Actuator Mass	Total Mass
Initial Design	1.0	1.0	1.0	1.0	1.0
Control-optimized Design $\beta = 0.15$	0.75	1.41	1.0	1.33	1.09
Integrated Design $\beta = 0.15$	0.32	4.82	0.58	1.97	0.97
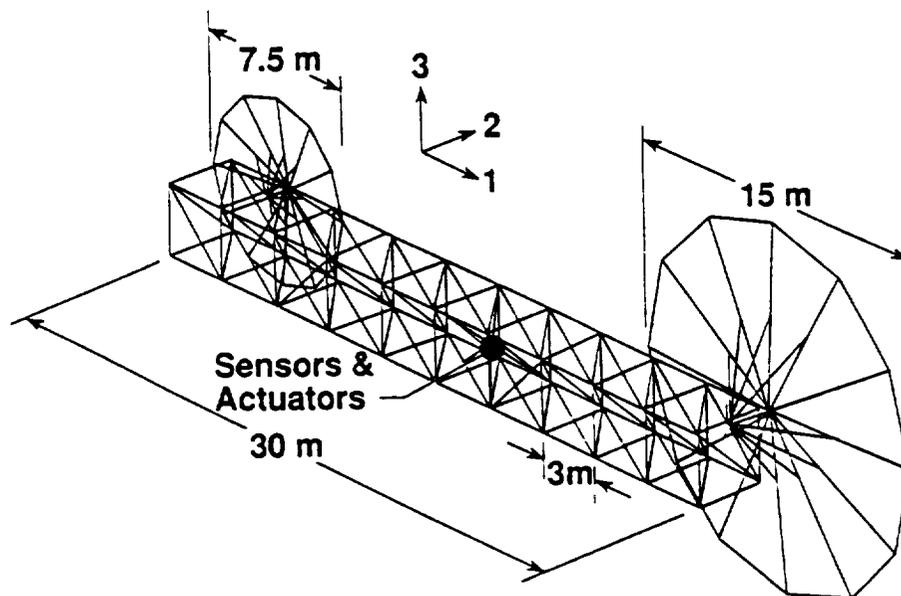
Figure 1. Generic geostationary platform.
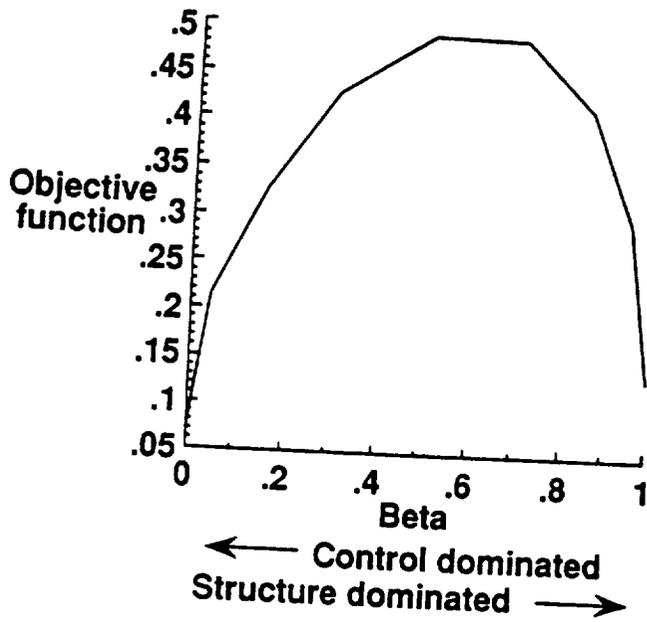
Figure 2
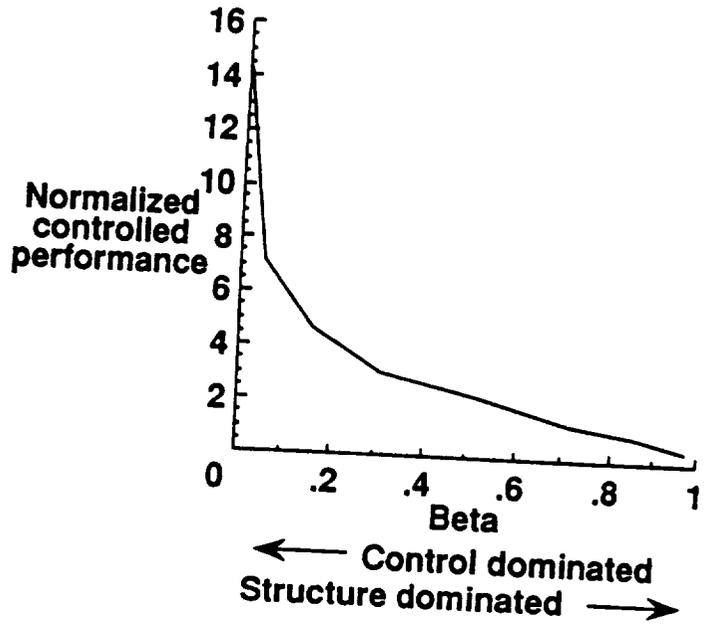
Figure 3
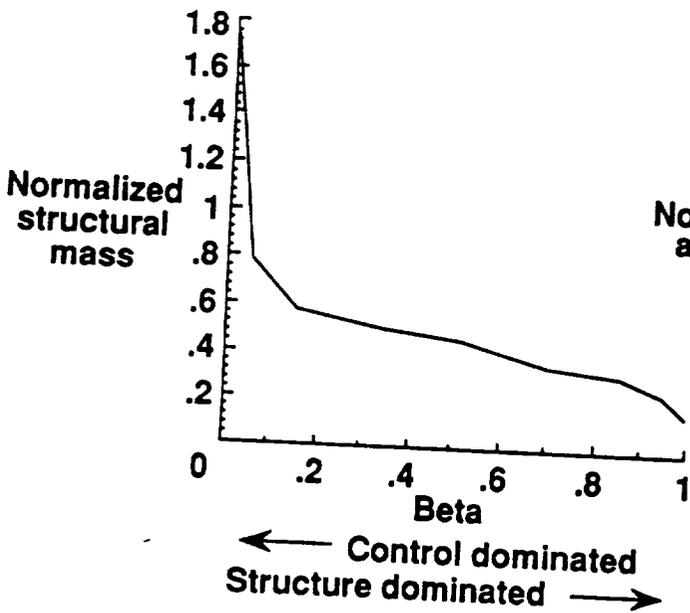
Figure 4
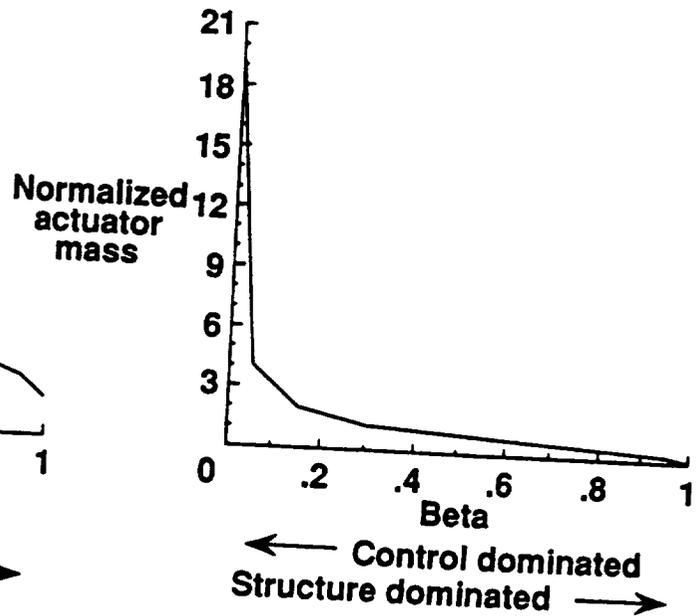
Figure 5

STRUCTURAL DAMPING OPTIMIZATION VIA

NON-OBSTRUCTIVE PARTICLE DAMPING TECHNIQUE

by

Hagop V. Panossian, Ph.D.
Principal Engineer, Control/Structure
ROCKWELL INTERNATIONAL CORPORATION
Rocketdyne Division
6633 Canoga Avenue
Canoga Park, CA 91303

Tel: (818) 710-6241

ABSTRACT

Presented herein is a novel passive vibration damping technique that is referred to as "Non-Obstructive Particle Damping (NOPD)." The NOPD technique consists of making small diameter holes (or cavities) at appropriate locations inside vibrating structures and filling these holes to appropriate levels with particles which yield the maximum damping effectiveness for the desired mode (or modes). Powders, spherical shaped, metallic, non-metallic or liquid particles (or mixtures) with different densities, viscosities and adhesive or cohesive characteristics can be used.

1. INTRODUCTION

The effectiveness of damping treatments in structures is related to the amount of vibration energy converted into other forms of energy. The performance of virtually all existing damping methods is affected by environmental conditions. Vibration damping under severe temperature, pressure, and fluid flow conditions is usually handled by structural design optimization, material selection, and other measures. Systematic treatments for passive damping are unavailable for cryogenic or harsh environments.

Existing Methods of Passive Damping

Presently used passive damping techniques can be classified into six broad categories: (a) viscoelastic material applications, (b) friction devices, (c) impact dampers, (d) fluid dampers, (e) tuned dampers, and (f) isolators.

a. Viscoelastic materials are very effective vibration suppressors at room and moderate temperatures but lose their effectiveness in low and high temperature environments. Viscoelastic materials have a tendency to degrade, embrittle, and even disintegrate with time through outgassing and other processes.[1]

b. Friction dampers are useful in many applications including harsh environments such as rocket engine turbine blades. However, because the performance of friction dampers is a function of the tightness of fit and of thermal and environmental conditions,[2] the effectiveness of frictional and thermal forces often degrades due to changes in surface conditions.

c. Impact dampers are used in applications where pyroshock conditions prevail, such as in recoil guns, and are relatively effective. Their effectiveness is attributed mainly to momentum exchange between the moving parts and impacting particles.[3] Impact dampers become impractical when the amount of energy to be absorbed is large.

d. Fluid dampers are devices that use the added mass effect, squeeze film effects, and where applicable, the sloshing effects of the fluid to enhance structural damping. They are also used to absorb sudden shocks by dissipating energy through heat and acoustic effects, and can be tuned to specific frequencies.[4] Fluid dampers are normally not applicable under harsh environmental conditions and hence have specialized and limited utility.

e. Tuned dampers form a class of vibration absorbers that have to be specially designed to attenuate the amplitude of a specific vibration mode. These include dynamic vibration absorbers, acoustic cavities, and other forms of passive tuned mechanisms. They are quite effective if designed appropriately. Sometimes, however, changes that occur in time can result in the detuning of the dampers.[5]

f. Isolators are damping devices designed to attenuate the transmission of vibrations to sensitive instrumentation and equipment over a given frequency range. These can be made from viscoelastic materials, fluid dampers, NOPD, and other combinations thereof. Their effectiveness is a function of vibration amplitudes, frequency bandwidth, and environmental conditions.[6] Isolators can also be affected by constant loading and vibration effects and can degrade in time.

2. **Non-Obstuctive Particle Damping (NOPD) Related Experience**

NOPD involves the potential of energy absorption/dissipation through friction, momentum exchange between moving particles and vibrating walls, heat, and viscous and shear deformations. Initial NOPD test results substantiate the effectiveness of this damping technique.[7] However, the information available is insufficient to model, optimize, and predict its effectiveness on different applications.

The following describes a Space Shuttle Main Engine (SSME) liquid oxygen (LOX) inlet tee vibration problem addressed and test results obtained that indicate significant vibration reductions using NOPD techniques.

3. **SSME LOX Inlet Tee.** In an effort to reduce the high amplitudes of vibrations of an SSME component[7],[8] without changing its mass or performance characteristics, four 1-millimeter (mm) diameter holes were machined inside one of the LOX inlet tee splitter vanes (Figure 1). The holes were partially filled with various particles and tested under hammer impacts and high frequency/high amplitude shaker excitations. Acceleration measurements were taken on the vane and on the outside shell of the LOX inlet tee (Figure 2) with holes empty and filled with various particles. The results showed significant effects in spite of the small size of the holes and small amounts of fill materials.

Description of Tests. Vibration and modal tests of the LOX inlet tee vanes, with holes empty, were conducted and data was recorded and reduced to the form of frequency response functions. Then, the four 1-mm-diameter holes were filled with 0.18 mm, 0.28 mm, and 0.58 mm (Figure 1) diameter steel balls to 1/2-, 3/4-, and 7/8-full levels and tested. Next, zirconium oxide (ZrO_2) ceramic balls of 0.25 mm diameter were introduced into the holes and tested for vibration levels with the same excitation. Similar tests were carried out with nickel and tungsten powders.

All of the tests were performed according to the standard practice of modal/vibration testing. The tee was suspended by flexible rubber bands to simulate a free-free condition, and the shaker was bolted onto a fixture with the moving tip (with a load cell attached to it) glued to the bottom of the tee as shown in Figure 2. The driving point response was kept at 13.7 g, and the vane responses at different locations along the midspan ranged from 20 g to 154.6 g at the leading edge midpoint of the right vane.

SSME LOX Inlet Tee Test Results. Two types of data were obtained in the tests: modal data (mode shapes and damping ratios at various frequencies); and vibration (accelerance levels of various modes with different types and levels of fill.

The vibration mode shapes of the vanes were obtained from a 25-point uniform grid of acceleration measurements on each vane. These mode shapes are shown on separate plots (Figure 3). As the summed frequency response function indicates (Figure 4), there are approximately 10 modes between 3000 and 6000 Hz and only a few below 3000 Hz. The dominant modes are above 4200 Hz and are torsional with the strongest at 4740 Hz. The less prominent modes below 4200 Hz are bending modes (Figure 3).

The damping ratios of these modes were quite low. They ranged from 0.06% for a strong symmetrical torsional mode at 4748 Hz to 0.20% for a mode at 5239 Hz. Eight of the 10 modes above 3000 Hz were isolated with the accelerance (acceleration per unit force) amplitudes and damping ratios for each material fill (at 3/4-full) recorded. The changes in damping performance with so little mass added (the mass of steel removed was approximately 1 gram and the amount of the heaviest material (tungsten) added was also approximately 1 gram) are remarkable. The amplitude reduction with tungsten was generally the greatest (Table 1). For example, for the torsional mode at 5021 Hz, the damping ratio was 0.0006 and the amplitude was 52.8 g/lb when empty. It changed to 0.0035 and 9.5 g/lb (Figure 5), respectively, when filled with tungsten. This illustrates a damping effect exceeding a factor of 5. A sample of three modes is presented in Figures 5 through 7. Also, it was found experimentally that the damping characteristics of the undrilled LOX inlet tee splitter vane were essentially the same as the drilled one with empty holes.

4. **Heuristic Evaluations.** The modal and vibration tests described contribute to the knowledge base of damping treatments. To the best of the author's knowledge, the methodology presented here is new. The observation that tiny amounts of various materials added to such a small volume at selected points produces such a large effect is novel. The potential application of such an approach to rocket engine components, spacecraft, aircraft, rotorcraft, lasers, and many other structural systems, is promising. Further research is necessary to fully understand the mechanisms

involved, optimal fill levels, and the best locations and dimensions of holes for specific applications as well as other aspects of NOPD.

As the data in Table 1 indicates, the damping effectiveness is a function of mass/density with other parameters probably playing equal parts. For instance, 0.18 mm diameter steel shot performs better than other particles for the modes at 3807 Hz and 4309 Hz frequencies, while nickel powder damps more than other particles at 4257 Hz frequency. Similarly, ZrO_2 was found to reduce vibration amplitudes in the above mentioned tests more than nickel or tungsten powders at 4309 Hz and 5239 Hz frequencies. However, in most cases, tungsten performs better than the rest. These facts indicate that vibration reduction by NOPD is a complex function of the material and size of particles relative to the cavities they are in. Hole diameter, density, and perhaps other characteristics of each type of the particles related to adhesive and cohesive forces, viscosity, friction surfaces and flexural properties, also contribute to the overall effect.

The NOPD technique is proven to be a very effective vibration damping methodology that has potential applications in all areas of structural vibration and acoustics. The tests presented herein show effectiveness in the high frequency range, but preliminary test data has indicated effectiveness under low frequency vibrations as well. Moreover, the NOPD concept is simple, easy to implement (holes can be made a part of the manufacturing process) and is relatively inexpensive. It has advantages over viscoelastic damping, since its effectiveness is independent of the environment (when appropriate particles are used), has more mechanisms for energy dissipation, does not add mass (it often can actually reduce mass) and does not degrade in time (among others). Furthermore, damping can be optimized through experiment and analysis by choosing the right location and size of holes in a structure, and by determining the optimal size-shape-kind (or even mixture) and the fill/compaction of the particles utilized.

5. REFERENCES

1. A.D. Nashif, D.I.G. Jones, and J.P. Henderson, <u>Vibration Damping</u>, John Wiley & Sons, N.Y. 1985

2. A.V. Srinivasan, "Dynamic Friction" in <u>Large Space Structures: Dynamics and Control</u>, S.N. Alturi and A.K. Amos (Eds), Springer-Verlag, Berlin, pp. 179-194, 1988.

3. S.E. Semergil, N.Popplewell, and R. Tye, "Impact Damping of Random Vibration," <u>Journal of Sound and Vibration</u>, V.121, pp. 178-184, 1988.

4. Von A. Trochidis, "Vibration Damping due to Air or Liquid Layers" (in German), <u>Acustica</u>, V.51, pp. 201-212, 1982.

5. J.B. Hunt, <u>Dynamic Vibration Absorbers</u>, Mechanical Engineering Publications, London, 1979.

6. P.Santini, <u>Damping Effects in Aerospace Structures</u>, NATO AGARD-CP-277, proceedings of the 48th meeting of the AGARD Structures and Materials Panel, Williamsburg, VA, 2-3 April 1979.

7. H.V. Panossian, "Non-Obstructive Damping Applications for Cryogenic Environments," presented in the Damping '89 Conference, in Florida, February 1989.

8. G.M. O'Connor and J. Jones, "Flow-Induced Vibrations of the SSME LOX Inlet Tee Vanes," presented at the AIAA 24th Joint Propulsion Conference, Boston, MA, July 1988.

Table I. Amplitudes and Damping Ratios of LOX Inlet Tee Splitter Vanes
Under Various Materials Damping

Mode Frequency, Amplitude, Damping Reduction Factor	Holes Filled With Different Materials – 3/4 Full							
	Empty	Steel 23	Zirconium Oxide	Steel 7	Steel 11	Nickel Powder	Tungsten Powder	
Frequency (Hz) B Amplitude (g/lb) Damping ratio Vibration Reduction factor	3,807 30.2 0.0009 —	3,805 30.5 0.0009 1	3,805 26.3 0.0009 1	3,807 24.5 0.001 1.2	3,805 27.0 0.0009 1	3,807 29.3 0.001 1.2	3,804 27.5 0.0011 1.25	M O D E 1
Frequency (Hz) B Amplitude (g/lb) Damping ratio Reduction factor	4,064 57.5 0.0009 —	4,063 43.4 0.0011 1.2	4,061 39.8 0.0012 1.3	4,061 37.3 0.0013 1.4	4,060 34.9 0.0014 1.6	4,057 29.0 0.0017 1.9	4,056 25.2 0.0016 1.8	M O D E 2
Frequency (Hz) T Amplitude (g/lb) Damping ratio Reduction factor	4,257 27.6 0.0015 —	4,258 32.6 0.0011 -1.2	4,256 27.1 0.0015 1	4,259 30 0.0012 -1.1	4,257 30.5 0.0012 -1.1	4,257 20.4 0.0013 1.4	4,258 25.5 0.0013 1.1	M O D E 3
Frequency (Hz) T Amplitude (g/lb) Damping ratio Reduction factor	4,309 55.5 0.0012 —	4,308 48.5 0.0013 1.14	4,308 40.5 0.0013 1.4	4,308 52.8 0.0013 1.06	4,306 38.4 0.0014 1.45	4,306 46.4 0.0016 1.2	4,306 41.5 0.0015 1.34	M O D E 4
Frequency (Hz) T Amplitude (g/lb) Damping ratio Reduction factor	4,748 70.1 0.0008 —	4,744 49.0 0.001 1.4	4,743 42.7 0.0009 1.64	4,741 41.1 0.001 1.7	4,740 37.0 0.0013 1.9	4,737 35.0 0.0017 2	4,734 18.2 0.0028 3.9	M O D E 5
Frequency (Hz) T Amplitude (g/lb) Damping ratio Reduction factor	5,021 52.8 0.0006 —	5,017 30.1 0.0009 1.76	5,018 27.6 0.001 1.9	5,015 20.4 0.0012 2.6	5,014 18.9 0.0015 2.8	5,010 17.1 0.0017 3.1	5,010 9.4 0.0035 5.6	M O D E 7
Frequency (Hz) T Amplitude (g/lb) Damping ratio Reduction factor	5,239 29.5 0.002 —	5,233 26.4 0.0028 1.12	5,234 26.3 0.0025 1.12	5,235 20.5 0.0034 1.44	5,232 22.6 0.0025 1.31	5,232 32.7 0.0016 -1.11	5,234 30.9 0.0017 -1.05	M O D E 6
Frequency (Hz) T Amplitude (g/lb) Damping ratio Reduction factor	5,606 7.9 0.001 —	5,604 7.0 0.00011 1.13	5,603 7.0 0.0011 1.13	5,605 6.0 0.0012 1.32	5,603 7.0 0.0011 1.13	5,593 6.9 0.001 1.15	5,593 6.4 0.001 1.2	M O D E 8

Note: B = Bending Mode
T = Torsional Mode
Reduction factor = Amplitude empty ÷ Amplitude filled

13
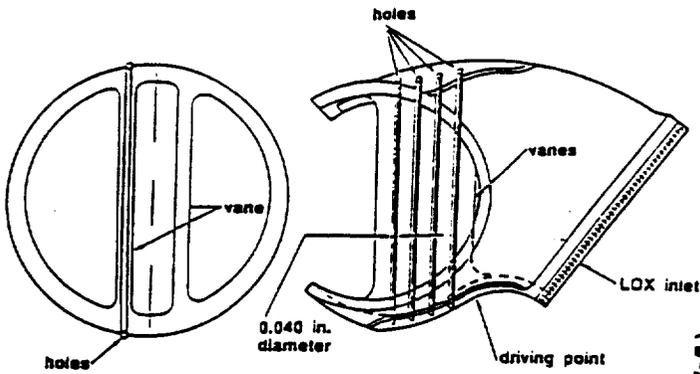
Figure 1. LOX Inlet Tee
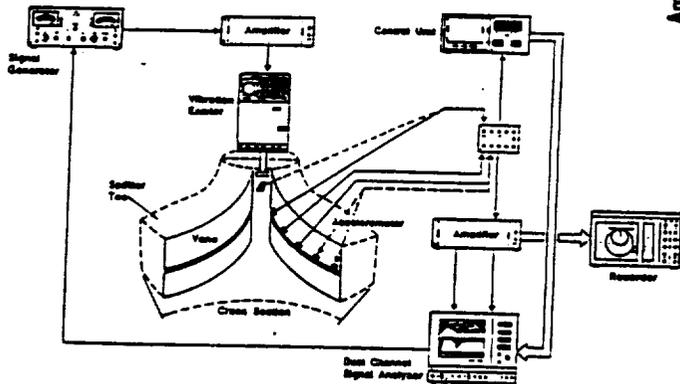
Figure 2. Measurements of Modal/Vibration Tests
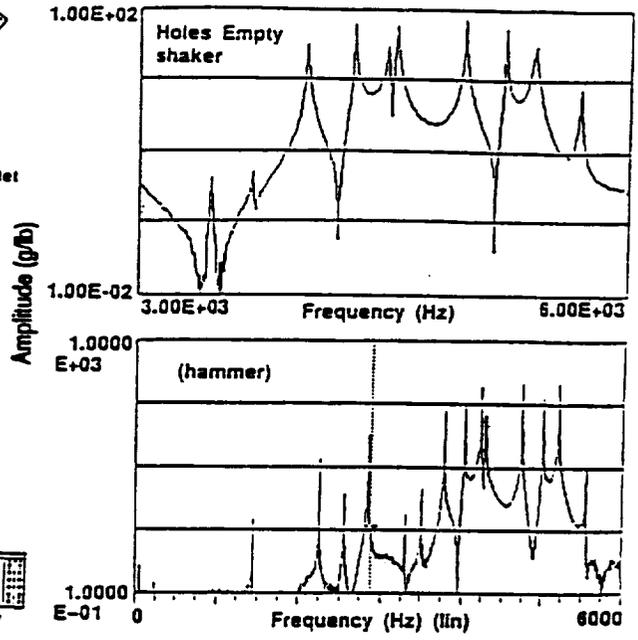
Figure 4. Vane Frequency Response Function

90PD-024-010
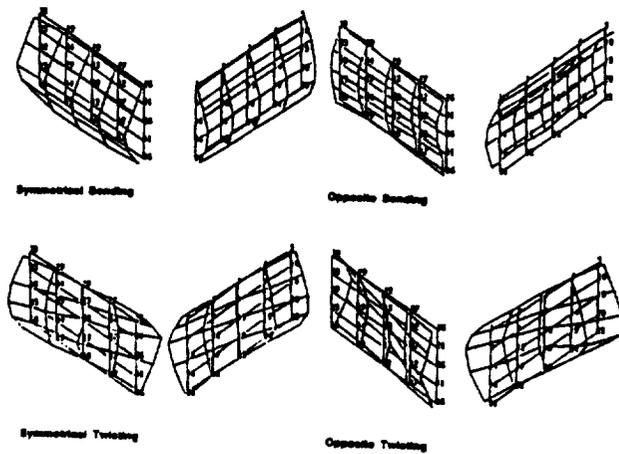
Figure 3. Vane Mode Shapes

14
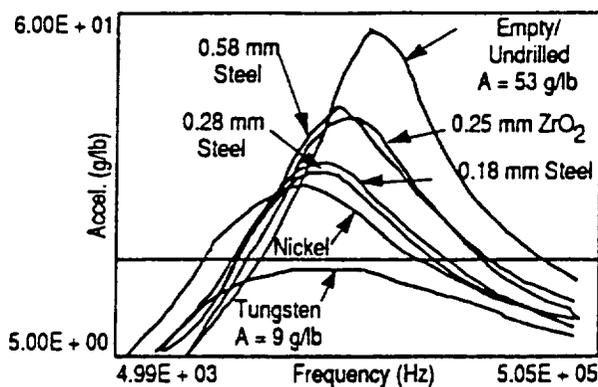
Figure 5. Accelerance Amplitudes of
a Torsional Mode at 5021 Hz Under
Various Materials
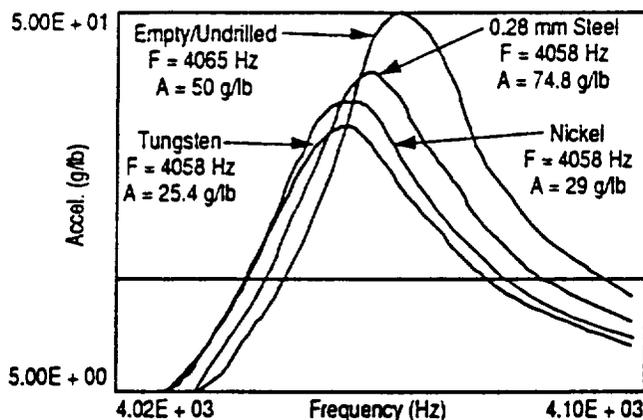
Figure 6. Accelerance Plots to Show Damp-
ing Effects of Various Materials for a Bending
Mode of the LOX Inlet Tee Splitter Vane
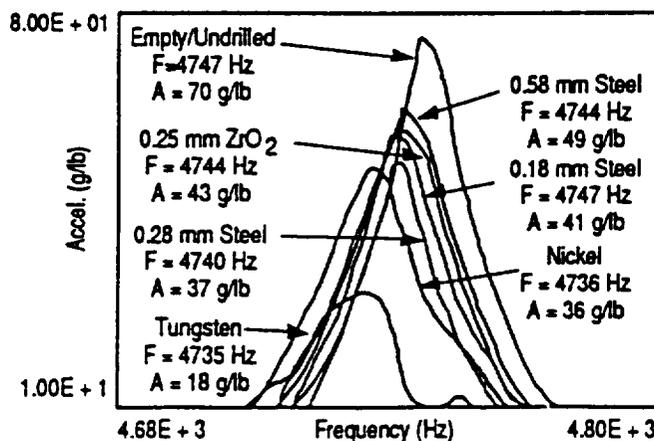
Figure 7. Accelerance Plots to Show Damp-
ing with Various Materials for a Torsional
Mode of the LOX Inlet Tee Splitter Vane

15

RELIABILITY OF DEGRADING DYNAMIC SYSTEMS SUBJECT TO DYNAMIC RANDOM LOADS

Mircea Grigoriu
Professor of Structural Engineering
Cornell University
Ithaca, New York   14853

Introduction

Consider the Daniels system in Fig. 1 with n brittle fibers of independent identically distributed resistances (R_i), i = 1, 2, ..., n, that is subject to a random load process $S(t)$. It survives in any of the damage states m = n, n-1, ..., 1 having m unfailed fibers and n-m failed fibers. System collapses when damage state m=0 is reached. Let Y_m be the residence period in damage state m. System reliability in a time interval $(0, \tau)$ is the probability

$$P_S(\tau) = P(\sum_{m=1}^{n} Y_m > \tau) \tag{1}$$

that time to failure $\sum_{m=1}^{n} Y_m$ exceeds service life τ.

Figure 2 shows a plate with a crack of initial length $2a_0$ that is excited dynamically by stresses $S(t)$, $t \geq 0$, normal to the crack. Let $A(t)$ be half crack length at time t. System reliability can be defined as probability

$$P_S(\tau) = P(A(\tau) < a_{cr}, 0 \leq t \leq \tau) \tag{2}$$

where a_{cr} is a specified critical crack length. The determination of this probability poses significant difficulties due to the coupling between system response and crack growth rate.

The paper develops methods for estimating system reliabilities $P_S(\tau)$ in Eqs. 1 and 2 and corresponding failure probabilities $P_F(\tau) = 1 - P_S(\tau)$. The analysis is based on properties of conditional differential random processes and one-dimensional diffusion models.

Daniels Systems

It is assumed that fibers are brittle linear elastic with stiffness K and damping C, unfailed fibers share equally the load, and system response is a mean square differentiable process that takes on positive values with nearly unit probability. Let $X_m(u)$ be system displacement in damage state m and u $\epsilon(0, Y_m)$ be a local time in this state. System response is

$$X_m(u) = \frac{1}{mk} S(Y_n + \ldots + Y_{m+1} + u) \tag{3}$$

for quasistatic excitations and satisfies the differential equation

$$\ddot{X}_m(u) + 2 \zeta_m \omega_m \dot{X}_m(u) + \omega_m^2 X_m(u) - \frac{1}{M} S(Y_n + \ldots + Y_{m+1} + u),$$

(4)

in which $\zeta_m - \sqrt{\frac{m}{n}} \zeta_n$, $\omega_m - \sqrt{\frac{m}{n}} \omega_n$, $\zeta_n - Cn/(2\omega_n M)$, $\omega_n - \sqrt{nK/M}$, and M — the system mass, for dynamic excitations.

Consider a sample $\hat{r}_1 < \hat{r}_2 < \ldots < \hat{r}_a$ of the ordered random resistances $\hat{R}_1 < \hat{R}_2 < \ldots < \hat{R}_a$ and denote by ξ_m the critical threshold in damage state m. Damage state m begins when $X_{m+1}(u)$ first upcrosses ξ_{m+1} and ends when $X_m(u)$ first upcrosses ξ_m. Probabilistic characteristics of $X_m(u)$ can be obtained from Eqs. 3 and 4 under initial conditions $X_m(0) - \xi_{m+1}$, $\dot{X}_m(0) - Z_{m+1} - \dot{X}_{m+1}(Y_{m+1}) | (a \ \xi_{m+1}$-upcrossing of $X_{m+1}(u)$ at time $u - Y_{m+1})$. It can be shown that Z_{m+1} follows probability (1)

$$f_{Z_{m+1}}(z \mid \xi_{m+1}) - \frac{z \ f(z \mid \xi_{m+1})}{\displaystyle\int_0^\infty dz \ z \ f(z \mid \xi_{m+1})}$$

(5)

where $f(z \mid \xi_{m+1})$ is the probability of $\dot{X}_{m+1}(Y_{m+1}) \mid X_{m+1}(Y_{m+1}) - \xi_{m+1}$.

Denote by $\upsilon_m(u)$ the mean ξ_m-upcrossing rate of $X_m(u) \mid (X_m(0) - \xi_{m+1}$, $\dot{X}_m(0) - z_{m+1} > 0)$ at time $u \geq 0$. Assuming that the sequence of ξ_m-upcrossings follows an inhomogeneous Poisson process of intensity $\upsilon_m(u)$, probability of event $(Y_m < y)$ given the above initial conditions on $X_m(u)$

can be approximated by $\exp \left(- \displaystyle\int_0^y \upsilon_m(u) \ du\right)$. The safety requirement in Eq. 1 becomes

$$\sum_{m-1}^{n} F_{Y_m}^{-1} \mid Y_n, \ldots, Y_{m+1}, Z_n, \ldots, Z_{m+1}, X_n(0), \dot{X}_n(0) \ (\Phi(U_{2(n-m)+3})) - \tau > 0$$

(6)

in which Φ — the distribution of the standard Gaussian variable, $U_{2(n-m)+3}$ — independent Gaussian variables with zero mean and unit variance, and functions F are distributions of conditional random variables $Y_m \mid Y_a, \ldots, Y_{m+1}, Z_a, \ldots, Z_{m+1}, X_a(0), X_a(0)$ where $(X_a(0), X_a(0))$ define the initial state of the system. First and second order reliability methods (FORM/SORM) discussed, e.g., in Ref. 2 can be used to calculate probabilities $P_S(\tau)$ and $P_F(\tau)$ based on the safety condition in Eq. 6.

Figure 3 shows the probability of failure for a Daniels system with n—2 fibers of deterministic strength $\hat{r}_1 - 1.25$ and $\hat{r}_2 - 3.00$ subject to a quasistatic load process $S(\tau) - d(\tau+\epsilon) S(\tau)$, where $\epsilon - 0.1$, $d(\alpha) - 1 - e^{-5}$, and $S(\tau)$ is a stationary Gaussian process with mean 2 and covariance function $(1 + |\tau_1 - \tau_2|) \exp(- |\tau_1 - \tau_2|)$. Results have been obtained by FORM/SORM algorithms applied to the safety condition expressed in the standard Gaussian space given in Eq. 6.

Plate with Crack

Consider the massless isotropic and homogeneous plate in Fig. 2 with an initial crack of length $2a_0$ that has an infinitely stiff element of mass M at its free end. The plate is subject to stresses S(t) normal to the crack. Let $2A(t)$, $X(t)$, and $g(A(t))$ be crack length, plate deformation, and plate stiffness at time $t \geq 0$. Displacement process X(t) satisfies differential equation

$$M \ddot{X}(t) + C \dot{X}(t) + g(A(t)) \, X(t) = S(t) \tag{7}$$

where C denotes system damping. Stiffness function g(a) can be obtained numerically for various values of the crack length and plate geometry. This function is approximated by

$$g(a)/g(0) = 1 - 1.708x^2 + 3.081x^4 - 7.036x^6 + 8.928x^8 - 4.266x^{10} \tag{8}$$

for $\ell = 1$ where $x = 2a/\ell$.

It is assumed that (i) S(t) is a stationary broad band Gaussian process, (ii) X(t) is positive with nearly unit probability, (iii) system is lightly damped, and (iv) crack growth is slow. Thus, the probability law of system response X(t) varies slowly in time so that it can be approximated by a narrow band Gaussian process with central frequency $\omega(A(t)) = \sqrt{g(A(t))/M}$. Let H(t) be the envelop of X(t) and $R(t) = H(t)/\sqrt{2} \, \alpha(A(t))$ where $\alpha(A(t))^2 \simeq \dfrac{\Pi \, G_0 \, M^2}{2 \, C \, g(A(t))}$ is the response variance at time t and G_0 is the coordinate of the one-sided power spectral density of S(t) at frequency $\omega(A(t))$. It can be shown by use of the averaging method that R(t) satisfies the stochastic differential equation (3)

$$dR(t) = - \rho \left[R(t) - \frac{1}{2R(t)} \right] dt + \sqrt{\rho} \, dB(t) \tag{9}$$

in which $\rho = C/2M$ and B(t) = the Brownian motion process with independent identically distributed Gaussian increments dB(t) of mean zero and variance dt. Therefore, the range of displacement process X(t) at time t can be approximated by $2H(t) = 2\sqrt{2} \, \alpha(A(t)) \, R(t)$.

According to the Paris and Erdogan model, the rate of crack growth is (2)

$$\frac{dA(t)}{dt} = \frac{\omega(A(t))}{2\pi} \, \alpha \, [\Delta\kappa(t)]^\beta \tag{10}$$

where α and β are coefficients and $\Delta\kappa(t)$ denotes the range of the stress intensity factor. Let h(a) be the stress intensity factor at the crack tip in Fig. 2 corresponding to unit stresses and a crack of length 2a. This function can be obtained numerically and is approximated by

$$h(a) = \sqrt{x} \, (0.467 - 0.514x + 0.960x^2 - 1.421x^3 + 0.782x^4) \tag{11}$$

for $\ell = 1$. Thus, stress intensity factor range $\Delta\kappa(t)$ is equal to h(A(t)) g(A(t)) 2 H(t). From Eqs. 9 and 10, process {R(t), A(t)} is a bivariate

diffusion process satisfying the stochastic differential equation

$$\begin{cases} dR(t) = -\rho \left[R(t) - \dfrac{1}{2R(t)} \right] dt + \sqrt{\rho}\ dB(t) \\[4mm] dA(t) = \eta\ w(A(t))\ R(t)^\beta\ dt \end{cases} \tag{12}$$

where $\eta = \dfrac{\alpha}{2\pi} \left[2M \sqrt{\dfrac{\pi G_0}{C}} \right]^\beta$ and $w(a) = \omega(a) \left[h(a) \sqrt{g(a)} \right]^\beta$.

These equations can be used to calculate the first passage time of $A(t)$ relative to level $a_{cr}(1)$.

An alternative approach can be based on the solution

$$A(t) = \psi^{-1} \left[\psi(a_0) + \int_0^t R(s)^\beta\ ds \right] \tag{13}$$

where $d\psi(a) = da/(\eta\ w(a))$. Since process $A(t)$ has almost surely monotonically increasing samples, reliability $P_S(\tau)$ is

$$P_S(\tau) = P\ (\bar{R}(\tau) < x_{cr}) \tag{14}$$

where $x_{cr} = \psi(a_{cr}) - \psi(a_0)$ and $\bar{R}(\tau) = \int_0^\tau R(s)^b\ ds$. Thus, $P_S(t)$ coincides with the distribution of random variable $\bar{R}(t)$ evaluated at x_{cr}.

An approximate method for calculating the distribution of $\bar{R}(\tau)$ can be based on the observation that random variables $R(t)$ and $R(s)$ are strongly or weakly correlated when $|t-s| < \tau_{cor}$ or $|t-s| > \tau_{cor}$, where τ_{cor} is the correlation time of $R(t)$. This suggests to approximate $R(t)$ by a stationary independent series with time step τ_{cor} and the same marginal distribution as $R(t)$ that takes on constant values within a time step. According to this approximation $\bar{R}(\tau)$ has mean

$$E\ \bar{R}(\tau) = n\ \tau_{cor}\Gamma \left[1 + \frac{\beta}{2} \right]\quad, \tag{15}$$

variance

$$Var\ \bar{R}(\tau) = n^2\ \tau_{cor} \left[\Gamma(1 + \beta) - \Gamma(1 + \frac{\beta}{2})^2 \right]\quad, \tag{16}$$

19

and characteristic function

$$\phi(u) = E[e^{iu\bar{R}(\tau)}] = [\bar{\phi}(u \ \tau_{cor})]^n \tag{17}$$

where $\tau = n \ \tau_{cor}$ and $\bar{\phi}$ is the characteristic function of random variable $R(t)^\beta$. Assuming that $\bar{R}(\tau)$ follows a Gaussian distribution reliability can be obtained from

$$P_S(\tau) = \Phi \left[\frac{x_{cr} - E \ \bar{R}(\tau)}{\sqrt{Var \ \bar{R}(\tau)}} \right] \tag{18}$$

where Φ = the distribution of the standard Gaussian variable. The Gaussian assumption is asymptotically correct as $\tau \to \infty$. Alternatively, the distribution of $\bar{R}(\tau)$ can be obtained from the characteristic function in Eq. 17 and the inversion theorem that gives (4)

$$F(x) = \frac{1}{2} + \frac{1}{2\pi} \int_0^\infty \frac{e^{iux} \phi(-u) - e^{-iux} \phi(u)}{iu} \ du \tag{19}$$

System reliability coincides with this distribution evaluated at $x = x_{cr}$, i.e.,

$$P_s(\tau) \simeq F(x_{cr}) \tag{20}$$

Figure 4 shows reliabilities in Eqs. 17 and 20 and Eq. 18 as a function of crack length $2a_{cr}$ for the plate in Fig. 3 with thickness of 0.1 in, $\ell = 1$ in, $M = 0.3$ ℓb sec^2 in^{-1}, c = 20 ℓb sec in^{-1}, a_0 = 0.05 in, G_0 = 0.0179 ℓb^2 sec in^{-4}, n = 100, τ_{cor} = 0.0479 sec, α = 0.66 x 10^{-6}, and β = 2.25. Results by the two approximate methods practically coincide.

Conclusions

Reliability has been determined for two degrading dynamic systems subject to random load processes. Damage is caused by loss of components for Daniels systems and crack extension for plates with cracks. The analysis has accounted for the coupling between response and current damage state of the system. It is based on mean crossing rates of conditional processes and properties of diffusion models. Simple systems are used to illustrate proposed methods for estimating reliability.

References

1. Grigoriu, M., "Reliability of Degrading Systems", EUROMECH 250: Nonlinear Systems Under Random Conditions, Como, Italy, June 19-23, 1989.

2. Madsen, H. O., Krenk, S., and Lind, N. C., Methods of Structural Safety, Princeton-Hall, Inc., Englewood Cliffs, New Jersey, 1986.

3.   Stratonovich, _Topics in Theory of Random Noise_, Vol. II, Gordon and
      Breach, New York, 1967.

4.   Gil-Pelaez, J., "Note on the Inversion Theorem," _Biometrika_, Vol. 28,
      Parts 3 and 4, Dec. 1951, pp. 481-482.
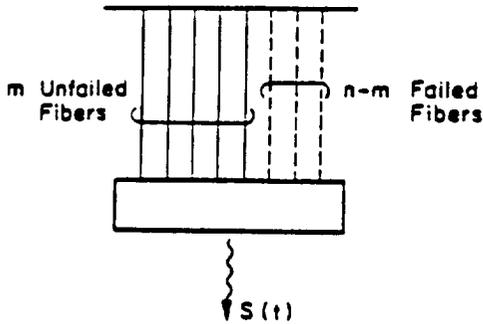
Figure 1. Daniels System with n Fibers in Damage State m ≤ n
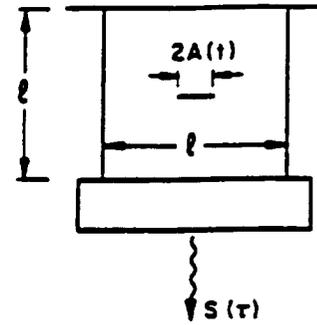
Figure 2. Plate with Initial Crack of Length $2a_0$
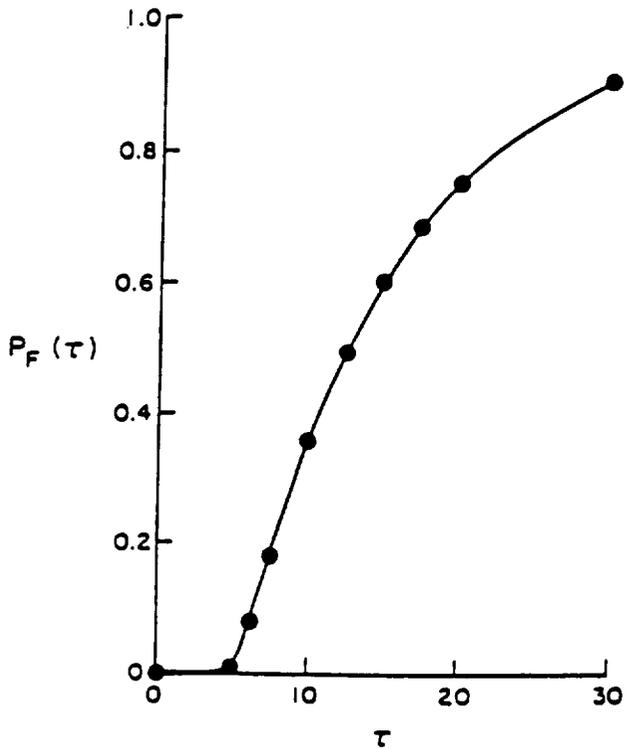
Figure 3. Probability of Failure $P_F(\tau)$ for a Daniels System with n = 2 Fibers.
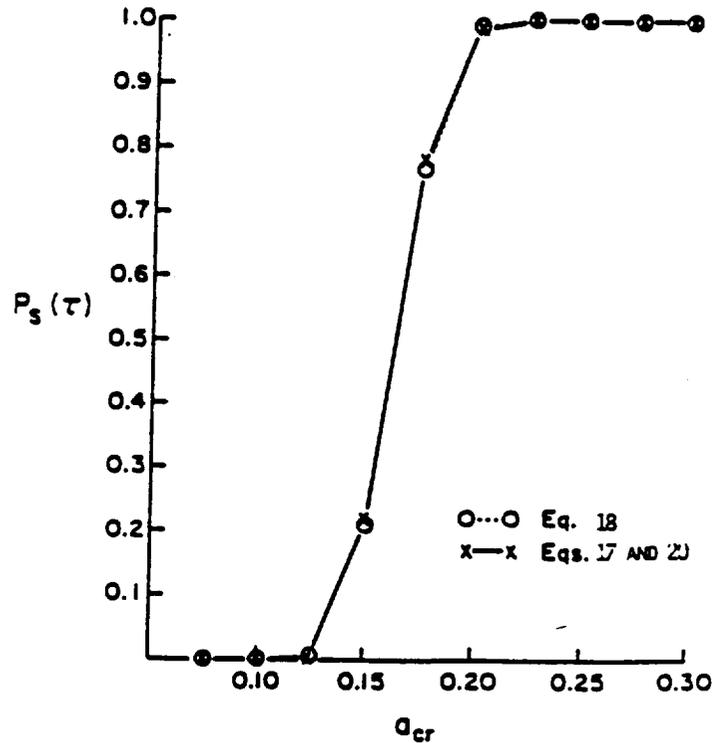
Figure 4. Reliability of Plate in Fig. 2.

21

COMPUTER SIMULATION OF MULTIRIGID BODY DYNAMICS AND CONTROL

M. Swaminadham*, Young I. Moon**, and V.B. Venkayya***
Wright Research and Development Center
Wright-Patterson Air Force Base, Ohio, 45433

ABSTRACT

The objective of this paper is to set up and analyze benchmark problems on multibody dynamics and to verify the predictions of two multibody computer simulation codes. TREETOPS and DISCOS have been used to run three example problems --- one degree-of-freedom spring mass dashpot system, an inverted pendulum system, and a triple pendulum. To study the dynamics and control interaction, an inverted planar pendulum with an external body force and a torsional control spring was modelled as a hinge connected two-rigid body system. TREETOPS and DISCOS affected the time history simulation of this problem. System state space variables and their time derivatives from two simulation codes were compared.

1.0 INTRODUCTION

Growing interest in deploying flexible satellite and spacecraft structures for various applications in space has made the subject of multibody dynamics important again. Rigid and flexible systems interconnected in either closed or open-loop configurations that undergo large rigid-body motions and/or small elastic deformations constitute the class of problems referred to as multibody dynamics problems. Prediction and control of systems for a combination of rigid and flexible-bodies motions are formidable tasks, but must be considered as part of the design strategy of multibody dynamic systems. The application of appropriate multibody dynamics analysis methods can achieve objective design of these systems.

The primary objective of this paper is to address Likin's question(1) on the absence of objective evaluation of government supported multibody computer codes. Two simulation codes --- TREETOPS and DISCOS were selected for evaluation. Three example problems were selected for analysis: they are a single degree-of-freedom spring mass system, a triple pendulum system, and an inverted pendulum with the base acted upon by a constant force. Time domain results for all three example problems obtained from both codes are discussed.

* M. Swaminadham, Presently Associate Professor, Texas A&M University,
** Young I. Moon, Senior Engineer, ASIAC,
*** V.B. Venkayya, Principal Scientist, Flight Dynamics Laboratory, WRDC

2.0 PAST RESEARCH

Two decades of research has produced a wealth of information about the dynamics of multibody systems. Of late, spacecraft components, large space structures and robotic manipulators with control elements were modeled as a combination of rigid and flexible bodies with an expanded generality to the system mathematical modeling and formulation of equations of motion.
Emphasis was given to the solution methods amenable to computers. Efficient dynamic simulation codes development was the major thrust.

Proceedings of the SDIO/NASA sponsored workshop on multibody simulation (2) provide important recent developments for computer simulations of rigid, flexible mulibody machine and space structures. Fletcher, et al (3) derived Newton-Euler equations of motion for two point-connected rigid bodies. Hooker and Magulies (4) formulated a generalized procedure for several multibodies.
Kane (5,6) introduced a new approach called Lagrange's form of D'Alembert's principle. This method contained the idea of generalized speeds for rapid computation of simulation problems. Meirovitch (7) derived the equations of motion of flexible spacecrafts and appendages by the conventional Lagrange's method.

To adapt mathematical methods for computer simulations, Fleischer (8) was the first to program the Hooker-Margulies equations. Several multibody simulation codes such as DISCOS (9), MBODY, TREETOPS (10A,10B), ADAMS (11), SADACS, MIADS, and CONTOPS have been developed. Some are designed for specific applications, while others claim to be general purpose programs. Kim and Haug (12) proposed a multibody dynamics verification library and presented results of DADS, DISCOS and CONTOPS.

3.0 MULTIBODY DYNAMICS COMPUTER PROGRAMS

Two computer codes specified for this study, DISCOS and TREETOPS, are briefly discussed in this section. Three example problems: (1) a spring-mass-damper system, (2) a tripe pendulum system, (3) an inverted pendulum system were selected and solved using each code.

3.1 TREETOPS

TREETOPS is a time history simulation of motion of a complex structure of interconnected flexible or rigid bodies at hinges. The equations of motion used by this code were derived via Kane's method, which is the generalization of Lagrange's form of D'Alembert's principle. In addition to geometry and material properties of the bodies of the structure, TREETOPS requires information such as the numerical integration type, time step size, plot-data output interval, simulation run time and other user supplied options.
The user can enter necessary data into the computer by running the interactive setup program TREESET, which acts as an interactive preprocessor to help the user enter and edit data in the various TREETOPS programs.

3.2 DISCOS

DISCOS (Dynamic Interaction Simulation of Controls and Structure) is another computer code written for the dynamic response analysis of topological systems of connected rigid bodies. It uses the general form of Lagrange's equations to derive the equations of motion. Lagrange multipliers are used as interaction forces and/or torques to maintain prescribed constraints.

DISCOS was used to study the stability of an inverted pendulum and to calculate the force which balances the system when the pendulum is released from a given initial position.

3.3 EXAMPLE PROBLEMS

3.3.1 Spring-Mass-Damper Problem

The first example is a single degree-of-freedom spring-mass-damper system, shown in Figure 1A. The oscillatory displacement of mass plotted against time is shown in Figure 2. Both TREETOPS and DISCOS predicted identical results.

3.3.2 Triple Pendulum Problem

The second problem consists of three pendulums (rigid bodies) connected at the hinged joints as shown in Figure 1B. The body 1 from its initial -30 degrees and bodies 2 and 3 from parallel positions were released. Figure 3 shows time history response of hinge 1, and other time histories of position and angular velocity are reported in Reference 13.

3.3.3 Inverted Pendulum Problem

Figure 1C shows the inverted pendulum mounted on a cart which is acted upon by a constant force of 10 Newtons. The cart is free to move in a horizontal direction unlike the previous two problems where translational motion of one hinge point was constrained. DISCOS and TREETOPS were used to obtained histories of positions and angular velocities of the hinge for the first 10 seconds. Details of these plots are shown in Reference 13. Figure 4 shows the hinge angle as a function of time.

3.3.4 Stability of Inverted Pendulum

The stability of the inverted pendulum was tried by feeding back a force on the cart. A constant row vector which is multiplied to state variables were determined from the characteristic values of the state differential equations. These nonlinear differential equations were solved numerically and plots were obtained for comparison with DISCOS predictions. The problem was run on DISCOS with 10 degree initial hinge angle displacement which was used to represent a kind of disturbances. Results show that the hinge angle starts to decrease as the computed force is applied to the cart and crosses the neutral line (upright position of the pendulum) twice before the system approaches the stable position asymtotically. DISCOS results are in good agreement with theoretical predictions as shown in Figure 5 and 6.

4.0 RESULTS AND DISCUSSION

From the results of all three example problems solved by both DISCOS and TREETOPS codes (i.e. time history plots of position, both linear and angular velocity, and acceleration), it can be concluded that both codes predict identical results. These identical simulation results support the conclusion in Reference 14, which states that all approaches used to derive equations of the motions will produce equivalent mathematical representatives. Reference 14 compares both these codes and presents capabilities and limitations. It also recommends a few baseline simulation tests using simplified and idealized configurations be conducted to eventually include actual configurations.

REFERENCES

1. Likins, P., "Multibody Dynamics - A Historical Perspective", Proceedings of the SDIO/NASA Workshop on Multibody Dynamics, September 1-4, 1987.

2. Proceedings of the SDIO/NASA Workshop on Multibody Dynamics, September 1-4, 1987.

3. H. Fletcher, L. Rongved and E. Yu, "Dynamic Analysis of A Two-Body Gravitationally Oriented Satellite", Bell System Technical Journal, Vol. 42, 1963.

4. W. Hooker and G. Margulies, "The Dynamical Attitude Equations for An N-Body Satellite", J. Astronaut. Sci., Vol. 12, No. 4, 1965, pp. 123-128.

5. T. Kane and C. Wang, "On the Derivation of Equations of Motion", J. Soc. Indust. Appl. Math., Vol. 13, No. 12, June 1985, pp. 487-492.

6. T. Kane and D. Levinson, "Formulation of Equations of Motion for Complex Spacecraft", J. Guidance and Control, Vol. 3, No. 2, March-April 1980.

7. L. Meirovitch and H. Nelson, "On the High-Spin Motion of A Satellite Containing Elastic Parts", Journal of Spacecraft and Rockets, Vol. 3, No. 11, November 1966, pp. 1597-1602.

8. G. Fleischer, "Multi-Rigid-Body Attitude Dynamics Simulation", JPL Technical Report 32-1516, February 15, 1971.

9. C. Bodley, A. Devers, A. Park and H. Frisch, "A Digital Computer Program for The Dynamic Interaction Simulation of Controls and Structure (DISCOS)", Vols. 1&2, NASA Technical Paper 1219, May 1978.

10A. User's Manual for TREETOPS -"A Control System Simulation for Structures with A Tree Topology", Dynacs Engineering Company, Clearwater, Florida, 1989.

10B. "TREETOPS Software Verification Manual", Dynacs Engineering Company, Clearwater, Florida, 1989.

11. ADAMS 5.2 User's Manual, Mechanical Dynamics, Inc., April 1987.

12. Kim, S. and Haug, E., "A Verification Library for Multibody Simulation Software", Proceedings of the Third Annual Conference on Aerospace Computational Control, JPL Publication 89-45, Vol. 2, Dec. 1989, pp. 917-928.

13. Swaminadham, M. and Moon, Y.,"Dynamics and Control of Multibody Dynamics", Report in preparation.

14. Lips, K. and Singh, R., "Obstacles to High Fidelity Multibody Dynamics Simulation", Proceedings of the 1988 American Control Conference, June 1988, Atlanta, Georgia, pp. 587-594.
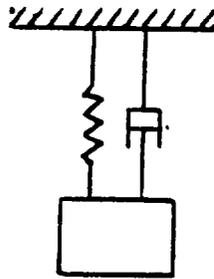
Fig. 1A   A Spring-Mass-Damper System
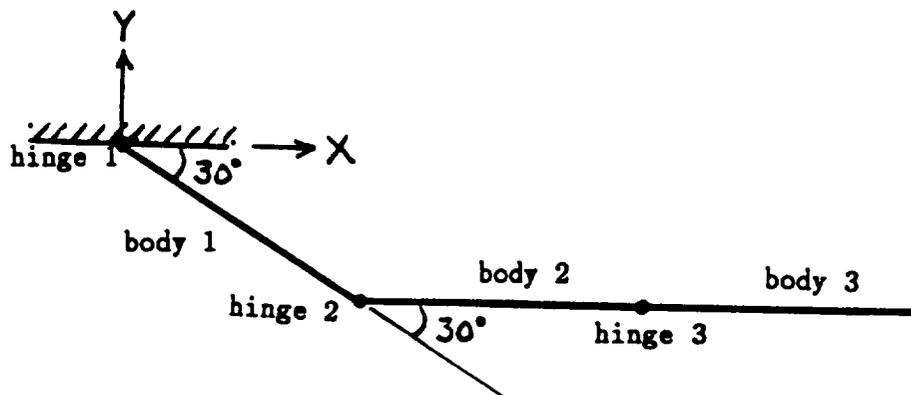
Fig.1B   A Triple Pendulum System
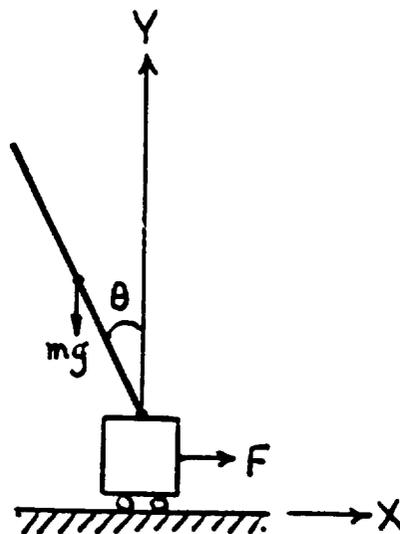
Fig.1C   An Inverted Pendulum System

Fig. 1   Three Example Problems
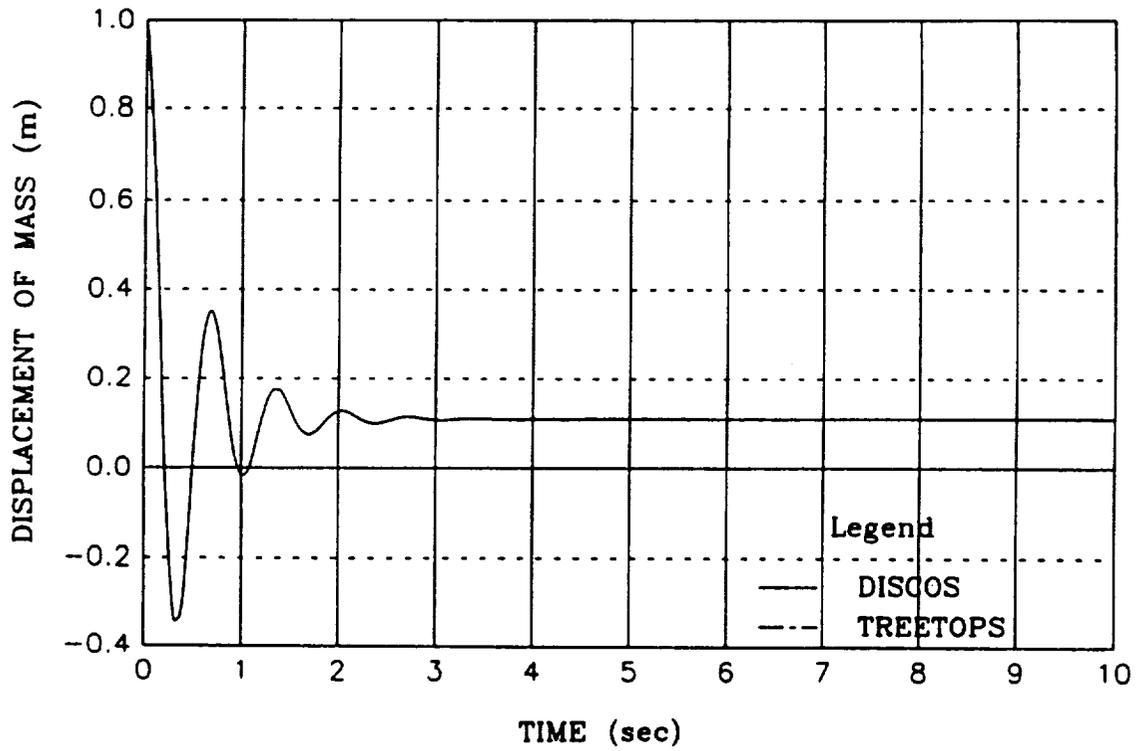
Fig. 2 Time History Response of Spring-Mass-Damper
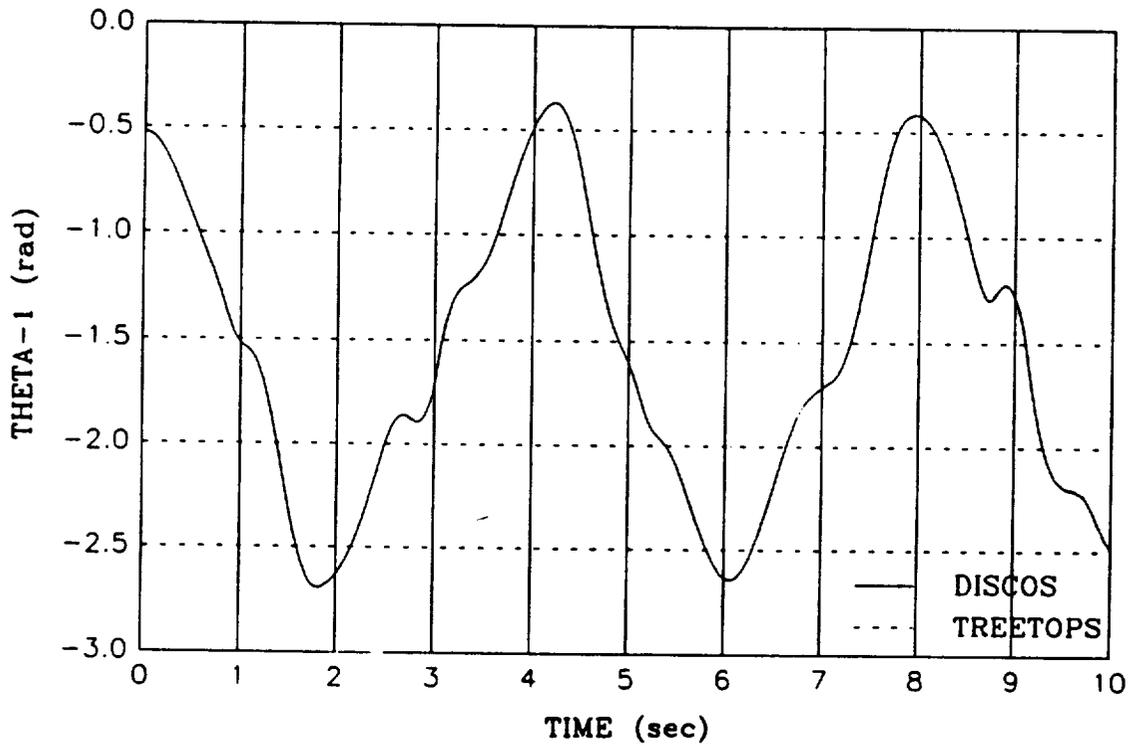
Fig. 3 Time History Response of Hinge Angle 1

28
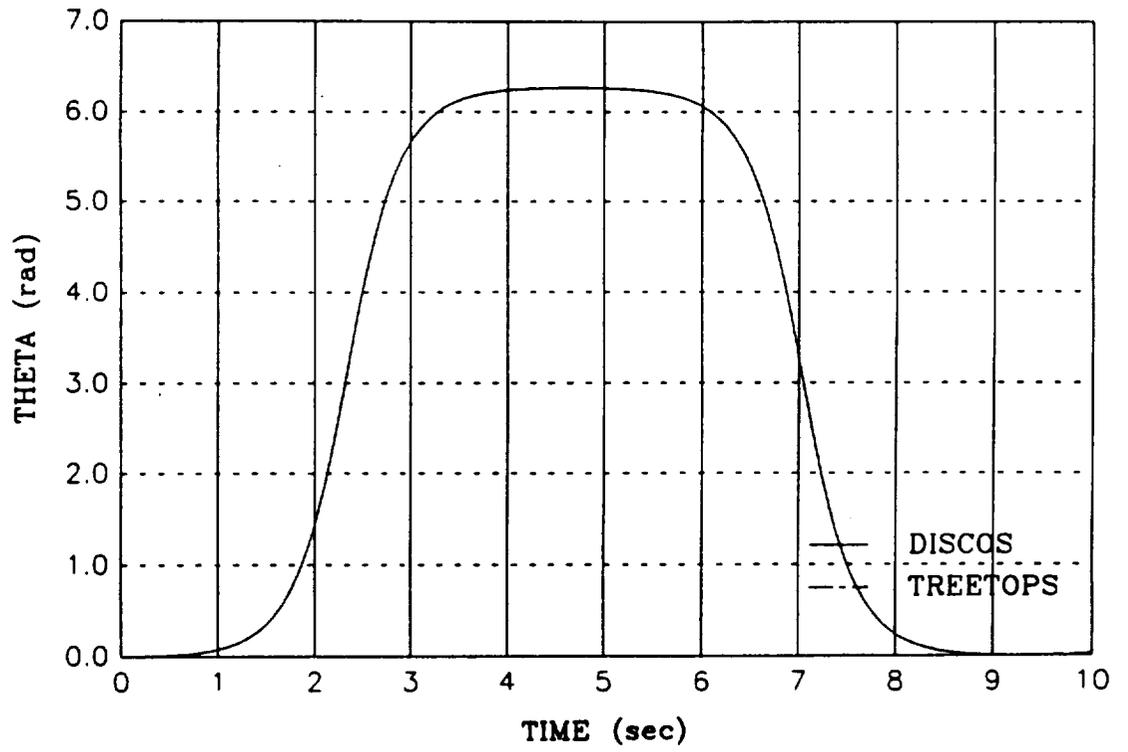
Fig. 4 Time History Response of Hinge - Inverted Pendulum
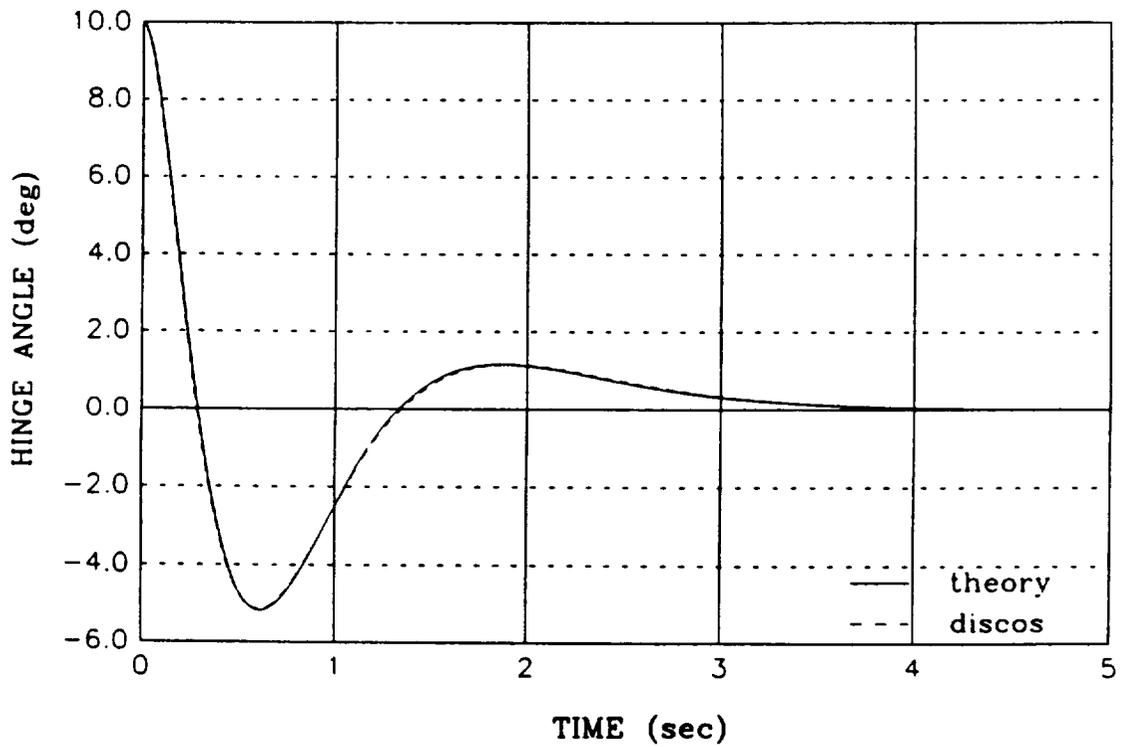
Fig. 5 Hinge Angle Response of Stabilized Inverted Pendulum

New Evidence Favoring Multilevel Decomposition and Optimization

Sharon L. Padula[1]
NASA Langley Research Center
Hampton, VA

Debra A. Polignone[2]
University of Virginia
Charlottesville, VA

INTRODUCTION

The issue of the utility of multilevel decomposition and optimization remains controversial. To date, only the structural optimization community has actively developed and promoted multilevel optimization techniques. (See reference 1 for a thorough overview and discussion of existing techniques.) However, even this community acknowledges that multilevel optimization is ideally suited for a rather limited set of problems. Reference 1 warns that decomposition typically requires eliminating local variables by using global variables and that this in turn causes ill-conditioning of the multilevel optimization by adding equality constraints. The purpose of the present paper is to suggest a new multilevel optimization technique. This technique uses behavior variables, in addition to design variables and constraints, to decompose the problem. The new technique removes the need for equality constraints, simplifies the decomposition of the design problem, simplifies the programming task and improves the convergence speed of multilevel optimization compared to conventional optimization.

STATE-OF-THE-ART MULTILEVEL OPTIMIZATION

Multilevel optimization is illustrated by the schematic in figure 1. The figure represents a three level decomposition of a general optimization problem into subproblems such that each design variable is assigned to one and only one subproblem. Starting with the lowest level, each subproblem (e.g. sub 3-1) is optimized with respect to its subset of design variables, holding all other design variables fixed. The objective of all subproblems is to minimize constraint violation. At the lowest level, only local constraints must be considered. However, at the middle level (e.g. sub 2-2), design variables may not be changed in a way which would violate lower level constraints. Finally, the system level optimization minimizes the cost function (objective) without violating constraints in any subsystem. The whole process is repeated until the values of the cost function and constraint violations are acceptable.

The key to multilevel optimization implementation is efficient minimization of constraint violations in each subproblem. Figure 2 illustrates how an envelope function (Ω) is used to fit multiple constraint functions (g_i). This envelope or cumulative constraint function is defined as:

$$\Omega = \frac{1}{\rho} \ln \left[\sum_i e^{\rho g} \right] \qquad (1)$$

where ρ is an adjustable smoothing factor. Each subproblem minimizes $\Omega(p,y)$ where p is a vector of fixed parameters and y is a vector of local design variables. The subproblem also calculates sensitivity derivatives $(\partial\Omega/\partial p_j)$ at the solution point.

---

1. Research Engineer
2. Graduate Student

In the problem illustrated by figure 1, the four subproblems on the lowest level would minimize cumulative constraint functions $\Omega_1, \Omega_2, \Omega_3$ and Ω_4. At the next level, the third subproblem (sub 2-3) treats Ω_3 and Ω_4 as if these were local constraints. The local design variables in sub 2-3 are some subset of the p_j which were fixed parameters influencing sub 3-3 and sub 3-4. Thus, the derivatives $(\partial\Omega/\partial p)$ calculated at the lowest level can be used to estimate new values of each Ω. At the system level, the actual cost function is minimized such that the constraints from all subproblems remain feasible $(\Omega_i < 0)$ or such that initially infeasible constraints do not get any worse.

Decomposition techniques

Multilevel optimization works very well for all system design problems which decompose readily into subsystems. Figure 3 illustrates the decomposition process. A grid of all system design variables (x) all subsystem design variables (y) and all constraints (g) is constructed. The grid is ordered with the system design variables first. The blocks indicate which constraints are functions of which design variables. In this simple example, it is easy to identify three subproblems with the first two constraints in sub 2-1, the next two in sub 2-2 and the final two in sub 3-1. Notice that sub 2-1 and 2-2 belong on the highest level of subsystems because they are functions of x. Sub 3-1 is on the level below that and connected to sub 2-1 by the first and second y design variables.

Unfortunately, not all system design problems are easy to decompose. Figure 4 illustrates a typical situation which occurs when some of the "fixed parameters" are not really independent design variables. Notice that in this problem, a fixed parameter in the lowest level, v_1, is not an independent design variable, but rather, a behavior variable which is a function of variables v_2 and y_3 at the middle level. Multilevel optimization is possible but the approximation of Ω at the middle level becomes more complicated. For instance,

$$\Omega_3(y_3 + \Delta y_3) = \Omega_3(y_3) + \{(\partial\Omega_3/\partial y_3) + (\partial\Omega_3/\partial v_1)(\partial v_1/\partial y_3)\} (\Delta y_3) \qquad (2)$$

If there are numerous behavior variables, then the programming logic required to resolve such approximations can quickly get out of control.

Novel Implementation technique

A novel way to simplify the coding is similar to the approach presented in reference 2. The dependency grid concept is extended to include behavior variables, v, and cumulative constraints, Ω, as shown in figure 5. This dependency grid is turned into a matrix, denoted Γ, by putting a 1 in each diagonal box, putting negative sensitivity derivative values in each shaded box and putting a zero in each blank box. For instance, a lightly shaded box in row v_1 and column y_3 is replaced by the most recent value of $(-\partial v_1/\partial y_3)$.

This global sensitivity matrix, Γ, is very useful. If a value for $(\partial\Omega_3/\partial y_3)$ is required, a column vector, β, which has zeros in each row except for a 1 in the row which corresponds to y_3 is constructed. The solution vector, d, to the system of equations

$$[\Gamma]d = \beta \qquad (3)$$

will contain the total sensitivity derivatives with respect to y_3. For instance, the row of d which corresponds to Ω_3 will contain

$$(d\Omega_3/dy_3) = (\partial\Omega_3/\partial y_3) + (\partial\Omega_3/\partial v_1)(\partial v_1/\partial y_3) \qquad (4)$$

31

It is important to understand that the global sensitivity matrix, Γ, is updated as the multilevel optimization progresses and that equation 3 is solved by each subproblem above the lowest level. The matrix Γ is initially the identity matrix. Each subproblem produces sensitivity derivatives ($\partial\Omega/\partial p$) and stores these in the appropriate locations of Γ. Those subproblems which evaluate behavior variables also calculate and store derivatives of behavior variables with respect to local design variables. Equation 3 can be solved for many different right hand side vectors thereby providing high quality gradients of each Ω with respect to all local design variables.

A global sensitivity matrix can also be created and used in the conventional optimization process (i.e. with no decomposition into subproblems). In that case, the rows and columns are design variables, behavior variables and constraints. The global sensitivity derivatives (dg/dy) and (dg/dx) include the effect of any behavior variables and can be used by the nonlinear programming algorithm to estimate more accurate search directions.

RESULTS

Table 1 contains results for several test cases comparing conventional approach (including global sensitivities) and multilevel approach. These test cases are generated and optimized using an extension of the multilevel simulator reported in reference 3. The multilevel cases are run to convergence. The conventional approach begins with the same initial values of design variables and is terminated after using approximately the same number of function evaluations as required by the multilevel approach. Three test cases are reported here. As noted in table 1, these cases vary in the number of design variables, number of constraint functions and number of behavior variables. The initial conditions are also varied ($g \leq 0$ is defined as the feasible region). In each case, the quality of the multilevel solution far exceeds that of the conventional approach in terms of smaller objective and constraint values.

The comparison between multilevel and conventional approaches would be even more lopsided if table 1 compared the number of constraint function evaluations required to converge to a global minimum. The conventional approach tends to follow constraint boundaries and therefore converges very slowly as more and more constraints become active. Multilevel approach, on the other hand, minimizes the constraint violation for each subproblem. Thus, the system level optimization begins far from most constraint boundaries and has considerable freedom to set the system level design variables. This is one reason why the multilevel optimization converges much faster than the conventional approach.

Figures 6-8 contain detailed convergence histories of the three problems. Objective (Obj) and maximum constraint value (Gmax=$\max_i g_i$) are plotted against number of constraint function evaluations. Notice that the conventional approach performs well when the initial design is in the feasible domain (figure 8) and performs poorly when the initial guess is far from the feasible domain (figure 7). On the other hand, multilevel approach performs equally well from any starting point.

CONCLUDING REMARKS

In conclusion, multilevel optimization can be implemented using a global sensitivity matrix. This formulation is easy to code because each subproblem is very similar and because all the coupling information is preserved in the sensitivity matrix. This formulation extends the usefulness of multilevel optimization to a much wider range of multidisciplinary design problems because decomposition is simplified if behavior variables are allowed. This method is particularly well suited to problems with large numbers of design variables and with computationally expensive constraints. The recent tests suggest that multilevel optimization converges much more rapidly than the conventional approach for such problems.

REFERENCES

1. Haftka, R.T., Gurdal, Z. and Kamat, M.P.: Structural Optimization, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1990, Chapter 10.
2. Sobieszczanski-Sobieski, J.: Sensitivity of Complex, Internally Coupled Systems. AIAA Journal, Vol. 28, No. 1, p. 153-160, Jan. 1990.
3. Padula, S.L., and Sobieszczanski-Sobieski, J.: A Computer Simulator for Development of Engineering System Design Methodologies. NASA TM-89109, Feb. 1987.

A. Test Problem Characteristics			
Initial conditions	infeasible	infeasible	feasible
Design Variables	50	50	66
Constraints	33	33	46
Behavior Variables	0	4	10
B. Multilevel Results			
Obj	.51	.38	.15
Gmax	.05	.00	.00
C. Conventional Results			
Obj	.83	.54	.31
Gmax	.25	11.10	.00

Table 1. Comparison of test problem results.
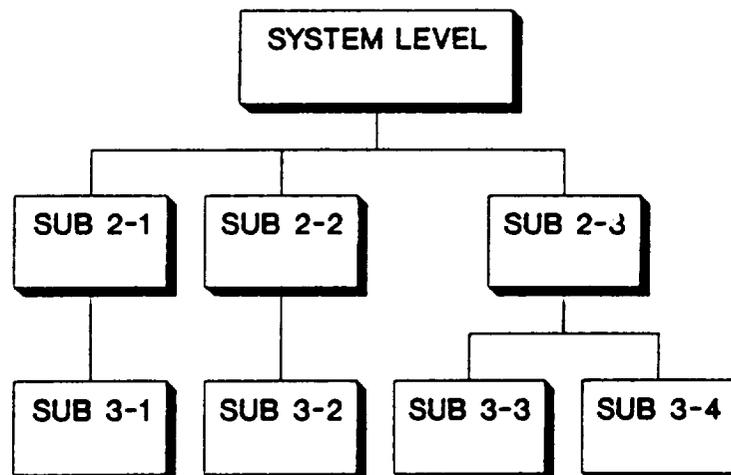
Figure 1. Schematic of a three-level optimization problem.
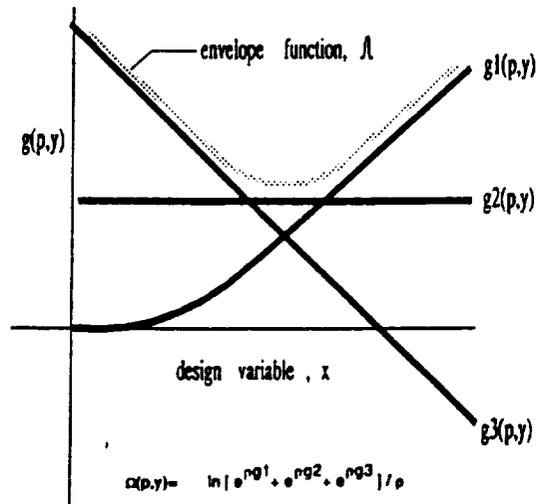
Figure 2. Envelope function used as cumulative constraint measure.
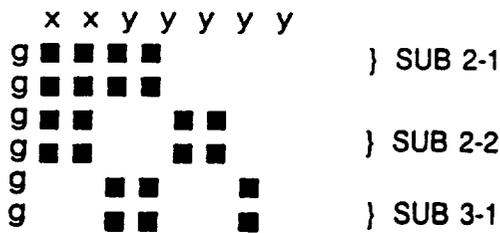
Figure 3. Grid of information used in decomposition process.
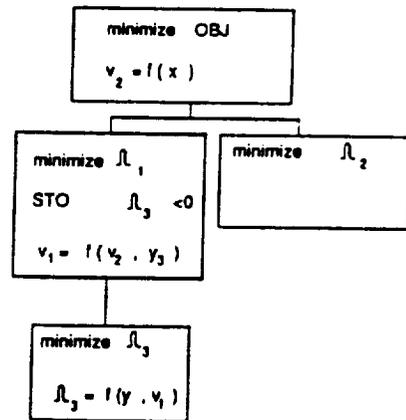
Figure 4. Multilevel decomposition with behavior variables.
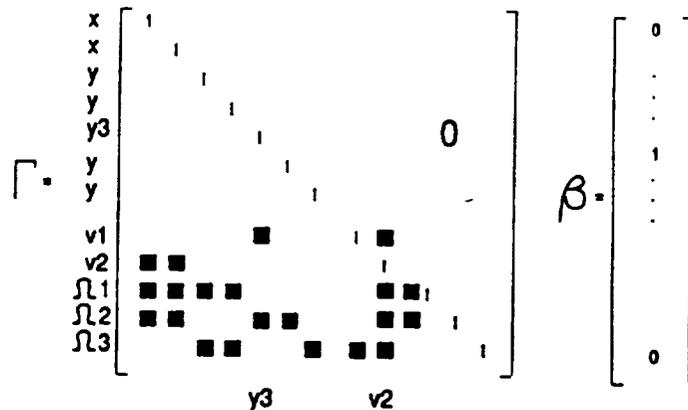
Figure 5. Global sensitivity matrix used for internally coupled systems.

Conventional Optimization
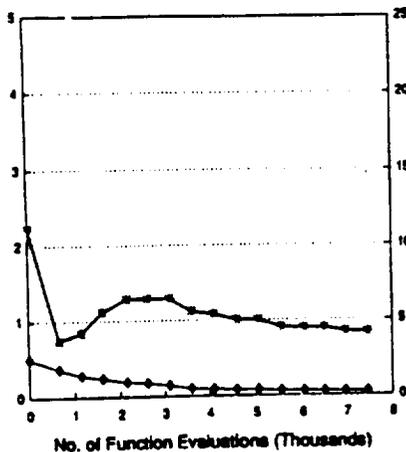
Multilevel Optimization
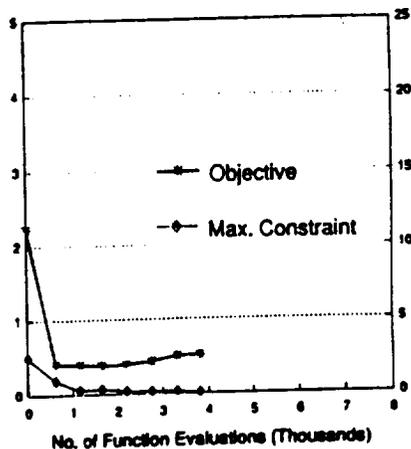
Figure 6. Comparison of optimization results (no behavior variables)

Conventional Optimization
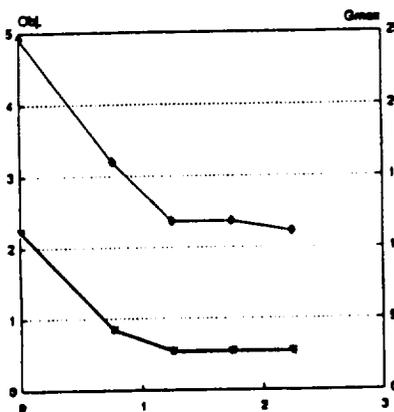
Multilevel Optimization
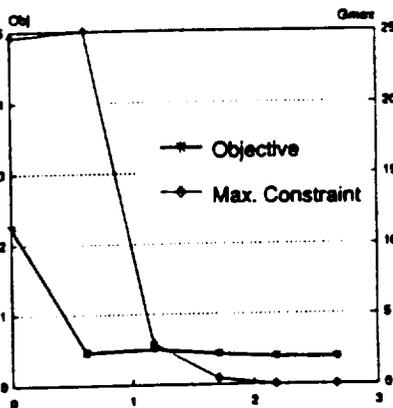
Figure 7. Comparison of optimization results (4 behavior variables)

Conventional Optimization
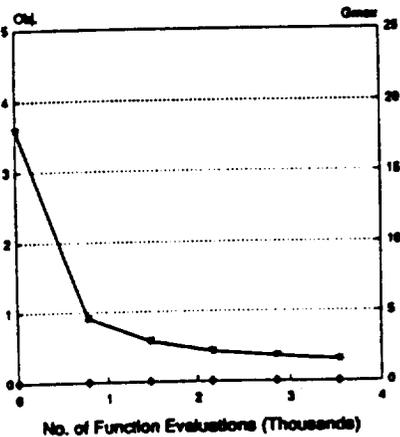
Multilevel Optimization
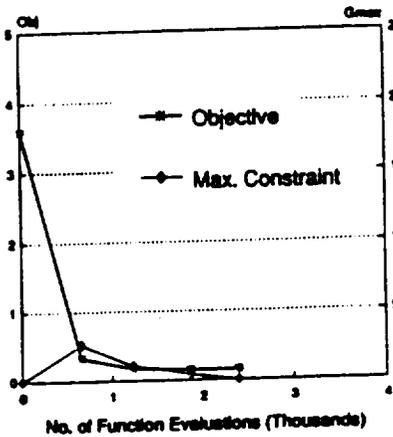
Figure 8. Comparison of optimization results (10 behavior variables)

DIRECT HANDLING OF EQUALITY CONSTRAINTS IN MULTILEVEL OPTIMIZATION

John E. Renaud
(Instructor)

Gary A. Gabriele
(Associate Professor)

Department of Mechanical Engineering, Aeronautical Engineering
and Mechanics
Rensselaer Polytechnic Institute
Troy, New York 12180-3590

ABSTRACT

In recent years there have been several hierarchic multilevel optimization algorithms proposed and implemented in design studies. Equality constraints are often imposed between levels in these multilevel optimizations to maintain system and subsystem variable continuity. Equality constraints of this nature will be referred to as *coupling* equality constraints. In many implementation studies these coupling equality constraints have been handled indirectly. This indirect handling has been accomplished using the coupling equality constraints' explicit functional relations to eliminate design variables (generally at the subsystem level), with the resulting optimization taking place in a reduced design space. In one multilevel optimization study where the coupling equality constraints were handled directly, the researchers encountered numerical difficulties which prevented their multilevel optimization from reaching the same minimum found in conventional single level solutions. The researchers did not explain the exact nature of the numerical difficulties other than to associate them with the direct handling of the coupling equality constraints.

In this paper, the coupling equality constraints are handled directly, by employing the Generalized Reduced Gradient (GRG) method as the optimizer within a multilevel linear decomposition scheme based on the Sobieski hierarchic algorithm. Two engineering design examples are solved using this approach. The results show that the direct handling of coupling equality constraints in a multilevel optimization does not introduce any problems when the GRG method is employed as the internal optimizer. The optimums achieved in this study are comparable to those achieved in single level solutions and in multilevel studies where the equality constraints have been handled indirectly.

INTRODUCTION

Recent studies (1-8) in the area of multilevel optimization have shown that it is a viable method for solving large non-linear design problems. In multilevel optimization, the main design problem at the top level is decomposed into a hierarchical tree consisting of subproblems at the lower levels. A coordination problem is introduced to preserve the coupling among these subproblems. The advantage of the decomposition is that it allows the subproblems to be analyzed and optimized independently with the coupling providing continuity between the levels.

In many multilevel optimization schemes one component of this coupling is achieved by placing equality constraints on various design variables between successive levels. In several implementation studies these equality constraints are eliminated explicitly and handled in some indirect fashion. Sobieski et al. (2,3) in their portal frame example implicitly enforced the equality constraints by eliminating variables at the lower levels. This is possible when an explicit relationship between local level and global level variables exists. Haftka (7) used two inequality constraints to replace the equality constraints in his multilevel approach to the same portal frame problem.

Thareja and Haftka (6) encountered numerical difficulties when handling equality constraints directly in a multilevel optimization. The numerical difficulties prevented their multilevel optimizing algorithm from reaching the same global minimum found in single level solutions. In an effort to avoid equality constraints Thareja and Haftka (8), have extended Haftka's earlier technique (7) in their single-level approach to hierarchical problems. The method provides a decoupling technique to form a single level problem which avoids equality constraints. The method is proposed for structural optimization. This decoupling may not always be an alternative in more complex engineering problems.

The objective of this paper is to demonstrate the utility of the Generalized Reduced Gradient (GRG) method for handling the coupling equality constraints directly in a multilevel optimization. The GRG method is used as the optimizer for both the system and subsystem optimizations within a multilevel optimization based on the Sobieski hierarchic algorithm (1-3). The coupling equality constraints imposed between system and subsystem variables are handled directly by the GRG optimizer. Two engineering design examples are solved, one being a two level study of the portal frame problem (2,7) and the second being a two level study of the speed reducer example (4,5,10-13). Numerical results are comparable with published results for both examples.

SOBIESKI HIERARCHIC ALGORITHM WITH THE GRG OPTIMIZER

Jaroslaw Sobieski of NASA Langley Research Center proposed a linear decomposition scheme (1) for hierarchic multilevel optimization in 1982. The algorithm has since been implemented in both two and three level design formulations (2,3). The algorithm requires an internal optimizer at both the system and subsystem levels. In the Sobieski test studies (2,3) the CONMIN (16) optimizer was employed as the internal optimizer, where the coupling equality constraints were enforced indirectly through variable elimination prior to optimization.

In this study OPT3.2, a FORTRAN implementation (14) of the GRG method is employed as the internal optimizer, with the coupling equality constraints being handled directly. In general a user can input equality constraints directly, when employing a GRG optimizer. In its optimization routine the GRG method divides the vector of design variables into two classes, nonbasic and basic variables, and employs the implicit function theorem to formulate a reduced, unconstrained

problem in the nonbasic variables. This problem is then optimized using a linear measure of the gradient of nonbasic variables (generalized reduced gradient). An important feature of the GRG method is that it can actively make variable basis changes when required during optimization. This differs from the fixed choice inherent to an explicit variable elimination.

A brief description of the Sobieski hierarchic algorithm for a two level formulation is presented in this section. In a two-level optimization, the original overall system design problem (standard form) is decomposed into a single problem at the system level and one subproblem for each element at the lower level. In general some of the system variables (global) are functions of the subsystem variables (local). At the system level only the global variables are used in the overall system optimization. A linear approximation of the subsystems (eq. 4) is monitored at the system level in order to preserve inherent coupling. The global variables being optimized at the system level serve as design parameters for the subsystems. Therefore the subsystem design space consists of only its own local variables, with the global variables fixed at the sublevel and acting as parameters.

At the subsystem the relationships between global and local variables are maintained and take the form of coupling equality constraints linking the variables (eq. 9). The constraints of the original problem are divided between the system and subsystems dictated in part by the variable restructuring (i.e., global vs. local). At the subsystem level these constraints are incorporated into an objective function referred to as a cumulative constraint function (eq 8). The goal in the subsystem problem is to minimize the cumulative constraint function in terms of the local variables for a given global variable state. After optimizing a given subsystem, parameter sensitivity derivatives (15) are calculated for both the local variables and cumulative constraint function. These sensitivity derivatives are used at the system level in a Taylor series extrapolation to form linear approximations of both the cumulative constraint and the local variables. System constraints are formed which insure that the linear approximations of the subsystem's cumulative constraint function and of the local variables do not exceed their respective bounds (eq 4,5). These linear approximations are of course functions of the global variables only. Move limits of $\pm 10\%$ are imposed on the global variables (eq. 6) to maintain the sensitivity derivative accuracy. The algorithm loops through the system and subsystems in an iterative fashion until a converged solution is obtained. The standard form of a two level optimization can be represented as detailed below:

SYSTEM LEVEL:

Minimize: $f(x)$ (1)

subject to:

$g^S_j(x) \geq 0$ $j=1,2,3,......J$ (2)

$h^S_k(x)=0$ $k=1,2,3,.....,K$ (3)

$(KS^O + (dKS/dx)^{T}* (\Delta x))_e \leq 0$ $e=1,2,..NE$ (4)

or achieve a 50% improvement if KS^O is positive (ref.17)

$y^l_e \leq [y^o_e + ((\Delta x)^{T}* (dY/dX)_e)^{T}] \leq y^u_e$ $e=1,2,..NE$ (5)

$ABS(\Delta x) \leq (0.10)* ABS(x^*)$ (6)

$x^l \leq x \leq x^u$ (7)

where:

$x=[x_1,x_2,x_3....x_n]^T$ vector of *global* variables

g^s system inequality constraint vector

h^s system equality constraint vector

KS   subsystem cumulative constraint function

$dKS/dx=[dKS/dx_1,........,dKS/dx_n]^T$

    vector of KS sensitivity derivatives

$y=[y_1,y_2,y_3....y_i]^T$ vector of *local* variables (subsystem)

$dY/dX=\{dY_j/dX_n\}$ (n x i) matrix of y sensitivity derivatives

$\Delta x=[\Delta x_1,\Delta x_2,.......\Delta x_n]^T$ vector of global variable change

subscript e   refers to specific subsystem element(s)
superscript o   refers to current value(s) at subsystem
superscripts l & u   refer to lower and upper limits
NE   total number of subsystems
superscript *   refers to initial values

SUBSYSTEM:

Minimize: $KS(x^*,y) = (1/p)*\ln[\sum_j e^{-pg_j(x^*,y)}]$ (8)

subject to:

$h_k(x^*,y)=(x^*-f(y)) = 0$ $k=1,2,....K_e$ (9)

where:

x^* *current* global variable vector (fixed)

$g_j(x^*,y)$ local inequality constraint $j=1,2,....J$

    normally stated as $g(x^*,y) \geq 0$

$h(x^*,y)$ coupling equality constraint vector

$KS(x^*,y)$ Kreisselmeier-Steinhauser function (11) used as a cumulative constraint function

p weighting factor in KS

subscript k   particular coupling equality constraint

K_e coupling equality constraints at subproblem "e"

38

DIRECT HANDLING OF COUPLING EQUALITY CONSTRAINTS

This research focuses on the direct handling of the equality constraints (eq. 9) which are used to maintain local variable (y) continuity with the current system level variable state (x^*). In the Sobieski implementation studies (2,3), these equality constraints have not been handled directly but are enforced implicitly, using variable elimination. This implicit enforcement results from using the coupling equality constraints to eliminate K_e local design variables by expressing them in terms of the fixed global variables. The resulting subsystem optimization then takes place in a reduced local design space without the equality constraints. This technique is not general and can only be used when an explicit relationship for variable elimination exists.

Although the Thareja and Haftka study (6) used a different multilevel optimization approach, we will discuss their work at this point. In their study, the subproblem optimization is based on minimizing the sum of the squares of the coupling equality constraints, where the coupling equality constraints have the same form as equation (9). Their subsystem inequality constraints are then imposed directly. They conclude that the direct use of these equality constraints introduces numerical difficulties which prevent a "true" optimum from being achieved. In fact their multilevel optimum was approximately fifty percent larger than the optimum found in single level solutions. Thareja and Haftka did not explain the exact nature of the numerical difficulties other than to associate them with the direct handling of the coupling equality constraints.

The Sobieski formulation at the subsystem (eq. 8,9) can be handled directly when using the GRG method. In this study the coupling equality constraints (eq. 9) are not eliminated, and no loss of optimality is observed. It should be noted that the GRG method does not require that the equality constraints be explicit functions. Therefore the use of the GRG method allows for the most general implementation of the Sobieski algorithm.

In the Sobieski algorithm, when equality constraints are handled directly (eq. 13), the optimizer must be able to handle infeasible starting points. This results from the fact that after a system level optimization is completed (new x^*), the subsystem coupling equality constraints (eq. 9) are likely to be in violation. Most GRG method implementations allow infeasible starting points.

In this study, the linear estimates of the local variables (eq. 5) are returned to the subproblems and used as the initial starting point for optimization. These linear approximations of local variables will tend toward satisfying the equality constraints and serve as an improved starting point for the subsystem optimization. Details of the GRG performance in finding a feasible starting point and of its optimizing performance are provided in the following engineering examples.

PORTAL FRAME EXAMPLE

The portal frame example which served as the test case for the Sobieski algorithm is recoded, and solved using the GRG code, OPT3.2 as the internal optimizer, and leaving the coupling equality constraints in explicit form. The portal frame shown in Figure 1 consists of three I-beams. The frame is designed subject to two loading conditions with the system level design problem being to minimize mass subject to frame displacement constraints. The global variables for each beam are the cross sectional area, "A" and area moment of inertia, "I." These terms are incorporated in the system level design vector x. For each individual beam (i.e., subproblem) the cumulative constraint function KS, is minimized (for local stress and buckling constraints), using the respective local design vector y, at each subsystem. These local variables, y are the cross sectional dimensions of each I-beam.

The coupling equality constraints imposed at the subproblems require that the current system values of area and inertia be maintained as explicit functions of the local y vectors. This constrains the y variables to those combinations which produce the current x values (i.e., area and inertia). Formal details of the deflection and local stress constraints imposed on the frame can be found in ref. 2. Additional details of this research's implementation of the Sobieski algorithm, with the GRG optimizer can be found in ref. 17.

The multilevel optimization of the portal frame using the GRG method as the internal optimizer produced numerical results which are improved, as compared to the Sobieski test case results. Table 1 details the optimized portal frame dimensions and system volume in comparison to Sobieski's results. It should be noted that the minimized volume achieved in this study is only slightly smaller than the result reported in the Sobieski test case. The differing optimums most likely represent two different local minimums for this highly non-linear problem.

Figure 3 details the convergence history of the portal frame's volume, where each node represents the system optimum as found by the GRG optimizer during iterations of the Sobieski algorithm. At convergence the algorithm tends to cycle about a portal frame volume of 90,000. cm^3 level. The minimum volume reported is simply the feasible minimum which occurs during these final cycles (see ref.17 for additional details on cycling of the Sobieski algorithm). Figure's 4a and 4b detail the GRG optimizer's typical iteration history at the subsystems in the feasible start trial. The GRG optimizer was able to fully optimize the subsystem problems taking on average only 6 GRG iterations per cycle of the Sobieski algorithm. An upper

limit of 12 GRG iterations was imposed at these subproblems and was reached in only a few cases. This upper limit prevented the true subsystem optimum from being reached for those few cases. This did not seem to impact the performance of the Sobieski algorithm.

As noted earlier, the initial starting point passed to the subsystems from the system is generally infeasible, excluding the original initialization. The infeasibility results from the coupling equality constraints being in violation. In this study the GRG optimizer, during its phase 1 search, was able to generate a feasible starting point for all but 4 of the 132 subsystem optimizations (i.e., the locations of the "zero" GRG iterations shown in figures 4). In these cases the last point generated in the phase 1 search is passed back to the system level along with limited sensitivity information. This approach proved adequate, as the GRG optimizer was able to generate a feasible starting point in the subsequent subsystem optimization.

The average of only 6 GRG iterations per subsystem optimization, along with the success of generating feasible starting points, can be considered "good performance," and clearly indicates the effectiveness of the GRG method in handling the coupling equality constraints imposed.

GEAR/SPEED REDUCER EXAMPLE

This example was originally modeled by Golinski (10,11) as a single level optimization. Several other optimization schemes have been applied to the problem including those by Lee (12) and Datseris (13). More recently Azarm and Li (4,5) solved the problem using a multilevel optimization scheme incorporating global monotonicity analysis. In their decomposition the variables being optimized at a given level are not explicitly related to those at other levels and therefore coupling equality constraints are not required. In complex large design problems it may not be possible to decompose problems such that variables are independent at different levels.

The Sobieski algorithm which allows for interdependent variables between levels is applied to the speed reducer problem in this study. The decomposition applied in this study involves variables which are explicitly linked between levels and therefore coupling equality constraints are imposed. These coupling equality constraints are enforced directly using the GRG optimizer.

Figure 5 shows a schematic of the speed reducer, which has been optimized for minimum mass subject to shaft deflection and stress constraints and to gear teeth stress/design constraints. The problem is decomposed, with the system level problem being to minimize mass subject to only shaft deflection and shaft stress constraints in a reduced design space. At the subsystem the gear stress/design constraints are minimized in a KS function while applying the coupling equality constraints to maintain variable continuity.

The global variables for the system are defined as follows:

$x_1 = l_1$ = shaft length 1 (between bearings)

$x_2 = l_2$ = shaft length 2 (between bearings)

$x_3 = d_1$ = shaft dia. 1

$x_4 = d_2$ = shaft dia. 2

x_5 = partial gear volume (explicit function of subsystem variables)

x_6 = transmitted gear force (explicit function of subsystem variables)

The variables at the subsystem or gear level are:

$y_1 = b$ = face width of the gear teeth

$y_2 = m$ = teeth module or the inverse of diametrical pitch

$y_3 = Z$ = number of pinion teeth

Where the coupling equality constraints for this formulation are;

$$x_5 = 0.7854 \, y_1 y_2^2 (3.333 \, y_3^2 + 14.9334 \, y_3 - 43.0932) - 1.5079 (x_3^2 + x_4^2) y_1 \qquad (11)$$

$$x_6 = 94000./(y_2 y_3) \qquad (12)$$

Note that the partial gear volume, x_5 which is dependent on all three subsystem variables (gear dimensions) is reduced to a single measure of that volume at the system level. The transmitted gear force, x_6 which depends on the gear dimensions is also reduced to a single variable at the system level. In the decomposition, volume (i.e., mass) is minimized at the system level using the shaft dimensions and x_5 (partial gear volume). This system level optimization is subject to shaft stress and deflection constraints, where the stress and deflection are caused by the transmitted force, x_6. At the subsystem (gear level) two gear stress constraints and three gear sizing constraints are incorporated into a KS function. The KS function is minimized (in y space) subject to two coupling equality constraints which restrict the y vector (gear dimensions) to combinations which maintain the current x_5 and x_6 values. Details of the decomposition used in this research can be found in ref. 17, where the constraint equations and system level functions are patterned after those formulated by Lee (12).

Figure 6 details the two level structure of the speed reducer optimization, where the decomposition breaks the design along the lines of shaft and gear design. Table 2 details the results of using the GRG method as internal optimizer within the Sobieski algorithm. One sees the results are similar to those found in the single level approach by Lee, the heuristic decomposition of Datseris's and the multilevel study of Azarm and Li's. Figure 7 details the convergence history of the speed reducer's volume during the multilevel optimization applied in this study.

The Sobieski hierarchic algorithm is based on monitoring linear measures of the subsystem at the system level. The linear approximations in this coding are based on sensitivity derivatives calculated using the Lagrange multiplier equations (15). The inputs to the Lagrange equations were calculated using analytic first and second order derivative information along with Lagrange multiplier estimates from OPT3.2. These analytic inputs result in highly accurate sensitivity derivatives. The impact of the accurate sensitivity derivatives can be seen in the GRG method's performance at the subsystem. Figure's 8a and 8b detail the iteration history of the GRG optimizer OPT3.2, at the subsystem for trials 1 and 2 respectively. In these plots the "zero" iteration points do not represent the GRG method's inability to generate a feasible starting point as in Figures 4 of the portal frame example. Instead "zero" iterations indicate that the linear extrapolations of the susbsystem variables returned from the system were feasible and optimal upon arrival to the subsystem. The Sobieski cycles requiring only one OPT3.2 iteration at the subsystem represent the case where the linear extrapolation returned from the system is infeasible, and the feasible point generated in the phase one search of OPT3.2 is optimal. Cycles requiring two or more OPT3.2 iterations represent an infeasible extrapolation from the system, followed by a phase one search, with a subsequent optimization using the GRG method. This example highlights both the robustness of the Sobieski hierarchic algorithm and the ability of the GRG method to handle coupling equality

CONCLUSIONS
Multilevel optimization methods are being considered for the design of increasingly complex systems. These multilevel methods decompose large design problems into a hierarchical organization of smaller subproblems. The subproblems can be optimized independently with a coordination problem being introduced to handle system coupling. In the most general decomposition it is likely that variables between levels will be functionally related. The multilevel optimization schemes based on handling this type of decomposition introduce coupling equality constraints to maintain the variable relationships. In some implementation studies these coupling equality constraints have been handled indirectly through variable elimination (2,3). This elimination is only possible when an explicit functional relationship between the variables exists. Another implementation study (7) suggests that the direct use of equality constraints may introduce numerical difficulties which prevent convergence of the optimizer. The ability to handle equality constraints directly, avoids the problem of variable elimination. In addition it also reduces the need to develop algorithms which avoid coupling equality constraints.

This research demonstrates that coupling equality constraints can be handled directly by using the GRG method as the internal optimizer in a multilevel optimization based on the Sobieski hierarchic algorithm. The two engineering design examples presented illustrate the GRG methods general utility for handling coupling equality constraints. These results are important in light of the fact that as multilevel optimization is applied to larger and more complex problems variable elimination may not always be possible. We should note that this success may not be exclusive to the GRG optimizer. Other optimizing algorithms which are robust in their handling of equality constraints may also work.

REFERENCES
1   Sobieszczanski-Sobieski, J., "A Linear Decomposition Method for Large Optimization Problems-Blueprint for development," NASA TM 83248, Feb. 1982
2   Sobieszczanski-Sobieski, J., James, B., and Dovi, A., "Structural Optimization by Multilevel Decomposition," AIAA Paper No 83-0832, AIAA/ASME/ASCE/AHS 24th Structures, Structural Dynamics and Materials Conference, Lake Tahoe, Nevada, May 1983.
3   Sobieszczanski-Sobieski, J., James, B., and Riley, M., "Structural Optimization by Generalized, Multilevel Decomposition," AIAA Journal, Vol. 25, Jan. 1987, pp. 139-145
4   Azarm, S., Li, W.-C.,1987, "Optimal Design Using a Two-Level Monotonicity-Based Decomposition," Advances in Design Automation," ASME Publication DE-Vol. 10-1, S.S. Rao (Ed.), pp. 41-48.
5   Azarm, S., Li, W.-C., "A Multi-Level Optimization-Based Design Procedure Using Global Monotonicity Analysis," Advances in Design Automation 1988," ASME Publication DE-Vol. 14, S.S. Rao (Ed.), pp. 115-120.
6   Thareja, R., Haftka, R., "Numerical Difficulties Associated with Using Equality Constraints to Achieve Multi-Level Decomposition in Structural Optimization," AIAA Paper No 86-0854-CP, AIAA/ASME/ASCE/AHS 27th Structures, Structural Dynamics and Materials Conference, San Antonio, Texas, May 1986.
7   Haftka, R. T., "An Improved Computational Approach for Multilevel Optimum Design," Journal of Structural Mechanics, Vol. 12(2), 1984, pp. 245-261.
8   Thareja, R., Haftka, R., "Efficient Single-Level Solution of Hierarchical Problems in Structural Optimization," AIAA Paper No 87-0716, AIAA/ASME/ASCE/AHS 28th Structures, Structural Dynamics and Materials Conference, Monterey, California, April 1987.
9   Kreisselmeier, G., Steinhauser, R., "Systematic Control Design by Optimizing a Vector Performance Index," IEAC Symposium on Computer Aided Design of Control Systems, Zurich, Switzerland, 1971.

10   Golinski, J., "Optimal Synthesis Problems Solved by means of Nonlinear Programming and Random Methods," Jnl. Mechanisms Vol. 5, pp. 287-309 (1970)

11   Golinski, J, "An Adaptive Optimization System Applied to Machine Synthesis," Mechanism and Machine Theory, 1973, Vol. 8, pp. 419-436. (1973).

12   Lee, T.W., "Weight Minimization of a Speed Reducer", ASME Paper No. 77-DET-163.

13   Datseris, P., "Weight Minimization of a Speed Reducer by Heuristic and Decomposition Techniques," Mechanism and Machine Theory, Vol. 17, No. 4, pp. 255-262, 1982.

14   Gabriele, G. A., Beltracchi, T. J., "OPT3.2: A FORTRAN Implementation of the Generalized Reduced Gradient Method," Users Manual, Department of Mechanical Engineering, Aerospace Engineering and Mechanics, Rensselaer Polytechnic Institute, Jan. 1988.

15   Sobieszczanski-Sobieski, J., Barthelemy, J.-F., and Riley, K.M., "Sensitivity of Optimum Solutions to Problem Parameters," AIAA Journal, Vol. 20, Sept. 1982, p. 1291.

16   Vanderplaats, G.N., "CONMIN-A Fortran Program for Constrained Function Minimization: User's Manual," NASA TM X-62282, Aug. 1973

17   Renaud, J.E., Gabriele, G.A., "Using the Generalized Reduced Gradient Method to Handle Equality Constraints Directly in a Multilevel Optimization," Advances in Design Automation 1989," ASME Publication DE-Vol. 19-2, B.Ravani (Ed.), pp. 7-14.

FIGURE 1: PORTAL FRAME DETAILS

FIGURE 1: PORTAL FRAME DETAILS

SECTION A-A

TYPICAL BEAM LOADING

TABLE 1

Comparison of multilevel optimization results
Portal Frame

	INFEASIBLE START			FEASIBLE START		
	Initial	(FINAL RESULTS)		Initial	(FINAL RESULTS)	
		Sobieski test case	ORG this study		Sobieski test case	ORG this study
Objective function (cm³)						
	26,469	92,090	87,595	275,000	92,330	89,653
Beam 1						
Y1	11.0	10.3	10.6	30.0	10.3	10.2
Y2	0.275	0.571	1.108	1.0	0.569	1.107
Y3	22.0	73.9	66.6	50.0	74.0	66.8
Y4	0.275	0.518	0.336	1.0	0.519	0.321
Y5	5.5	5.08	9.417	30.0	5.13	5.621
Y6	0.275	1.18	1.16	1.0	1.16	1.971
Beam 2						
Y1	11.0	10.3	10.0	30.0	10.7	10.0
Y2	0.275	0.476	0.895	1.0	0.451	0.721
Y3	22.0	89.4	76.9	50.0	90.1	79.5
Y4	0.275	0.414	0.327	1.0	0.417	0.337
Y5	5.5	5.14	5.84	30.0	5.12	6.757
Y6	0.275	0.984	1.15	1.0	0.960	0.982
Beam 3						
Y1	11.0	5.03	6.61	30.0	4.98	5.19
Y2	0.275	0.253	0.473	1.0	0.253	0.857
Y3	22.0	59.1	69.9	50.0	59.0	59.33
Y4	0.275	0.251	0.256	1.0	0.25	0.252
Y5	5.5	10.0	10.0	30.0	10.0	10.0
Y6	0.275	0.400	0.312	1.0	0.393	0.765

FIGURE 2: TWO LEVEL STRUCTURE

FIGURE 2: TWO LEVEL STRUCTURE

FIGURE 3: PORTAL FRAME CONVERGENCE

FIGURE 3: PORTAL FRAME CONVERGENCE

NOTE: The initial rise of the infeasible start case can be attributed to the 10% move limits imposed on global variables at the system level. These move limits artificially restrict the system level search for a feasible point.

FIGURE 4a: SUBSYSTEM 1 ITERATIONS

System-Subsystem Iterations
(Sobieski Algorithm)

FIGURE 4b: SUBSYSTEM 2 ITERATIONS

System-Subsystem Iterations
(Sobieski Algorithm)

FIGURE 5: SPEED REDUCER SCHEMATIC

TABLE 2

Comparison of optimization results; Speed Reducer

	LEE (single level)	DATSERIS (decomposition)	AZARM&LI (multilevel)	GRG(this study) (multilevel)
Y1	4.4	3.45	3.5	3.69
Y2	0.68	0.68	0.7	0.66
Y3	17.0	17.0	17.0	17.0
X1	6.93	6.93	7.3	6.60
X2	7.72	7.72	7.715	7.30
X3	3.35	3.35	3.35	3.35
X4	5.29	5.29	5.29	5.29
Objective function cm^3	2886.0	2867.4	2994.4	2851.0

FIGURE 6: TWO LEVEL STRUCTURE
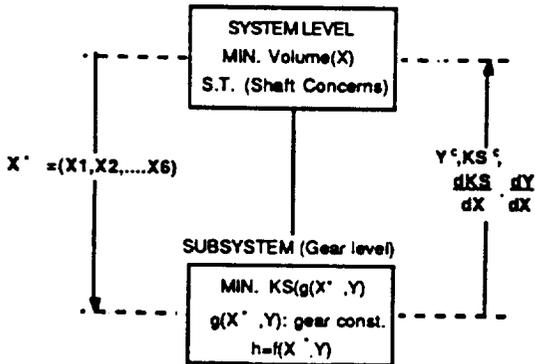
FIGURE 7: SPEED REDUCER CONVERGENCE

ITERATION (system-subsystem)
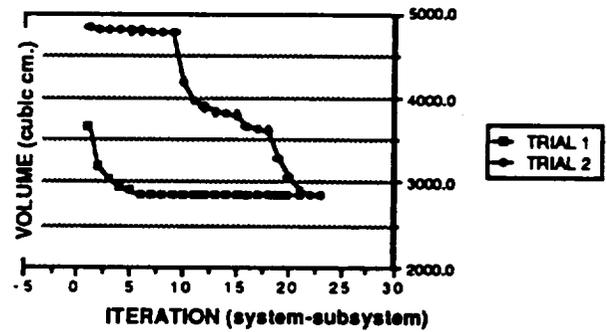
FIGURE 8a: GEAR LEVEL (TRIAL 1)

System-Subsystem Iterations
(Sobieski Algorithm)
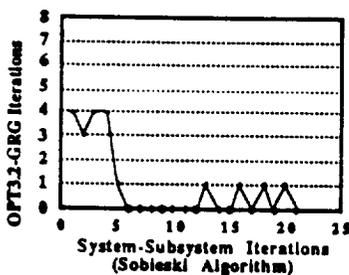
FIGURE 8b: GEAR LEVEL (TRIAL 2)

System-Subsystem Iterations
(Sobieski Algorithm)
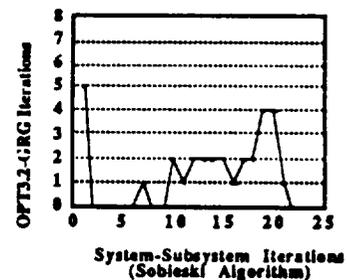
CONFIGURATION DESIGN SENSITIVITY ANALYSIS OF BUILT-UP STRUCTURES

Kyung K. Choi and Sung-Ling Twu

Department of Mechanical Engineering
and
Center for Simulation and Design Optimization
The University of Iowa
Iowa City, Iowa 52242

In the design of complex built-up structures that are made of truss, beam, membrane, shell, and solid, there are five different kinds of design variables: material property, sizing, shape, configuration, and topological variables. Previous research has shown that the improvement in performances obtained by altering the configuration of structural components can be much more significant than those obtained when the geometry is assumed to be fixed (Refs. 1~4). Using the variational approach, a unified design sensitivity has been developed in Ref. 5 for the first three kinds of design variables, and has been further extended recently in many structural analysis problem such as nonlinear, structural dynamics, and frequency response analysis (Refs. 6~8). In this paper, a continuum design sensitivity analysis method is developed for the configuration design variable of built-up structures.

One of the key differences between the shape and configuration design sensitivity analysis is the orientation change of the design component. In shape design problems, the domain shape is treated as the design variable, and the orientation of the design component remains fixed. On the other hand, in the configuration design, both the domain shape and the orientation of the design component are changed. The configuration design change of a design component can be viewed as a dynamic process of moving the design component in three steps: translation, rotation, and shape variation. These three steps are depicted in Fig. 1 for a line design component. Three similar steps can be applied to the configuration design change of the surface and solid design components. Since translation, rotation, and shape variation are three independent design changes, configuration design sensitivity can be obtained by adding the design sensitivity results that are obtained from each design perturbation.
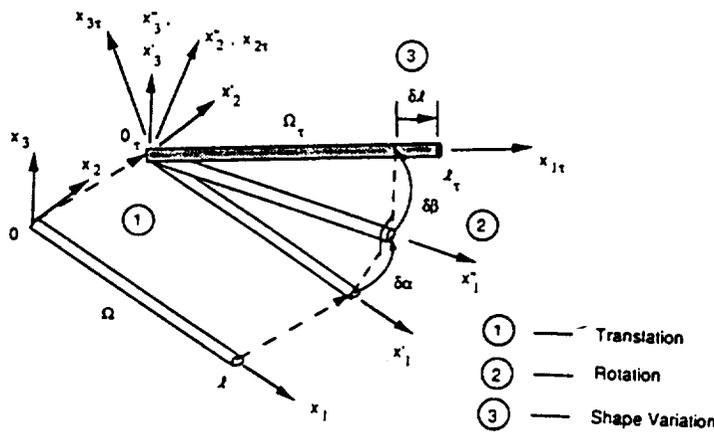
Figure 1 Configuration Design Change of a Line Design Component

It is shown in Ref. 9 that a translation of the design component does not contribute to the design sensitivity result of a performance measure. Therefore, the configuration design sensitivity result can be obtained by adding contributions from the shape variation and rotation of each individual design component in the built-up structure.

For shape variation, a unified shape design sensitivity analysis method has been developed in Ref. 5 using the material derivative idea of continuum mechanics. The domain shape variation can be viewed as a dynamic process of deforming a continuum medium from Ω to $\Omega_\tau = T_\Omega(\Omega,\tau)$, with τ playing the role of time. A shape design velocity field V_Ω is considered as the perturbation of the shape.

Suppose the displacement $z_\tau(x_\tau)$ is a smooth solution of the boundary value problem on the perturbed domain Ω_τ. The existence of pointwise material derivative $\dot{z}_{V_\Omega}$ at $x \in \Omega$ is shown in Ref. 5. If z_τ has a regular extension to a neighborhood U_τ of the closure Ω_τ, then the partial derivative z'_{V_Ω} exists and commutes with the derivative with respect to x_i as shown in Eq. (5). The pointwise material derivative of displacement is obtained in terms of the partial derivative and the shape design velocity field as shown in Eq. (4). Using the material derivative formulas of Ref. 5, the first variation of a general functional due to the domain shape variation is obtained in Eq. (7).

Material Derivative for Domain Shape Variation
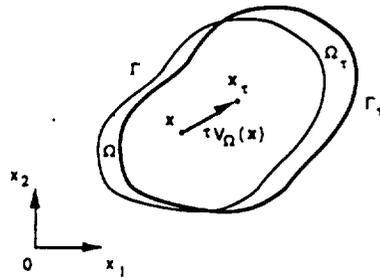
$$x_\tau = T_\Omega(x,\tau) = x + \tau V_\Omega(x) \\ \Omega_\tau = T_\Omega(\Omega,\tau)$$

(1)

$$V_\Omega(x,\tau) = \frac{dx_\tau}{d\tau} = \frac{\partial T_\Omega(x,\tau)}{\partial \tau}$$

(2)

Figure 2 Domain Shape Variation

$$\dot{z}_{V_\Omega}(x) = \frac{d}{d\tau} z_\tau(x + \tau V_\Omega(x)) \Big|_{\tau=0} = \lim_{\tau \to 0} \frac{z_\tau(x + \tau V_\Omega(x)) - z(x)}{\tau}$$

(3)

$$\dot{z}_{V_\Omega}(x) = z'_{V_\Omega}(x) + \nabla z^T V_\Omega(x)$$

(4)

$$\left(\frac{\partial z}{\partial x_i}\right)'_{V_\Omega} = \frac{\partial}{\partial x_i}(z'_{V_\Omega}), \quad \text{for } i = 1, 2, 3$$

(5)

$$\psi = \int_{\Omega_\tau} f_\tau(x_\tau)\, d\Omega_\tau$$

(6)

$$\psi'_{V_\Omega} = \int_{\Omega} [f'_{V_\Omega}(x) + (\nabla f^T V_\Omega) + f(\nabla^T V_\Omega)]\, d\Omega$$

(7)

Similar to the shape variation, the process of orientation change can be viewed as a dynamic process of rotating a continuum medium from Ω to $\Omega_\tau = T_\theta(\Omega,\tau)$, with τ playing the role of time. An orientation design velocity field V_θ is considered as the perturbation of the orientation, and is normal to the domain of the design component.

Suppose the displacement $z_\tau(x_\tau)$ is a smooth solution on the perturbed domain Ω_τ.

The pointwise derivative $\dot{z}_{V_\theta}$ at $x \in \Omega$ due to the orientation change, if it exist, is defined by Eq. (10), where, a regular extension of z_τ is defined as $z_\tau(x) \equiv z_\tau(x_\tau)$, if $x_\tau = x + \tau V_\theta$. As shown in Eq. (13), very much like z'_{V_Ω} in Eq.(5), z'_{V_θ} commutes with the derivative with respect to x_i. In Eq. (10), A is the rotational transformation matrix and $\tilde{V}_{V_\theta}$ contains derivatives of the orientation design velocity field, and they are written in Eqs. (11-12) for the line design component. The same derivation can be applied to a surface design component with different A and $\tilde{V}_{V_\theta}$. Using the regular extension of a displacement function, and the fact that the determinant of the Jacobin is independent of the orientation change, the first variation of a general functional due to the orientation change is obtained in Eq. (14).

Derivative for Orientation Design Sensitivity Analysis
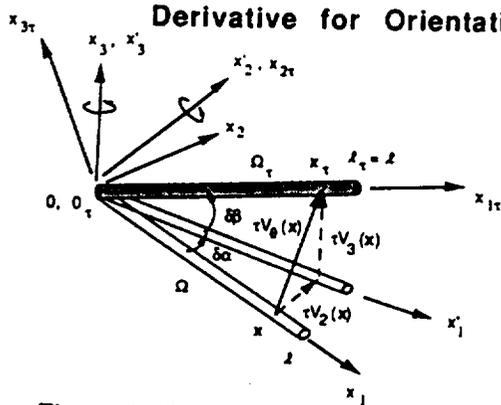
Figure 3 Orientation Change of a
Line Design Component

$$x_\tau = T_\theta(x,\tau) = x + \tau V_\theta(x)$$
$$\Omega_\tau = T_\theta(\Omega,\tau)$$
(8)

$$V_\theta(x,\tau) = \frac{dx_\tau}{d\tau} = \frac{dT_\theta(x,\tau)}{d\tau} = \frac{\partial T_\theta(x,\tau)}{\partial \tau}$$
(9)

$$\dot{z}_{V_\theta}(x) = \frac{d}{d\tau} z_\tau(x + \tau V_\theta(x)) \Big|_{\tau=0} = \lim_{\tau \to 0} \frac{z_\tau(x + \tau V_\theta(x)) - z(x)}{\tau}$$

$$= \lim_{\tau \to 0} \frac{z_\tau(x) - z(x)}{\tau} + \lim_{\tau \to 0} \frac{z_\tau(x + \tau V_\theta(x)) - z_\tau(x)}{\tau}$$

$$= z'_{V_\theta} + \lim_{\tau \to 0} \frac{[(A(\tau V_{2,1}, \tau V_{3,1}) - A(0, \tau V_{3,1})) + (A(0, \tau V_{3,1}) - I)] z_\tau(x)}{\tau}$$

$$= z'_{V_\theta} + \left[\frac{dA(\tau V_{2,1}, 0)}{d\tau} \Big|_{\tau=0} + \frac{dA(0, \tau V_{3,1})}{d\tau} \Big|_{\tau=0} \right] z(x)$$

$$= z'_{V_\theta} + \tilde{V}_\theta z(x)$$
(10)

$$\tilde{V}_\theta = \begin{bmatrix} 0 & -V_{2,1} & -V_{3,1} & 0 & 0 & 0 \\ V_{2,1} & 0 & 0 & V_{3,1} & 0 & 0 \\ V_{3,1} & 0 & 0 & V_{2,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & -V_{2,1} & -V_{3,1} \\ 0 & 0 & 0 & V_{2,1} & 0 & 0 \\ 0 & 0 & 0 & V_{3,1} & 0 & 0 \end{bmatrix}$$
(12)

$$\left(\frac{\partial z}{\partial x_i} \right)_{V_\theta} = \frac{\partial}{\partial x_i}(z'_{V_\theta}), \quad \text{for } i = 1, 2, 3$$
(13)

$$\dot{\psi}_{V_\theta} = \int_\Omega \dot{f}_{V_\theta}(x) \, d\Omega$$
(14)

$$A(\delta\alpha,\delta\beta) = \begin{bmatrix} \cos\delta\alpha & -\sin\delta\alpha & 0 & 0 & 0 & 0 \\ \sin\delta\alpha & \cos\delta\alpha & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \sin\delta\alpha & 0 & 0 \\ 0 & 0 & 0 & \cos\delta\alpha & -\sin\delta\alpha & 0 \\ 0 & 0 & 0 & \sin\delta\alpha & \cos\delta\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\delta\beta & 0 & -\sin\delta\beta & 0 & 0 & 0 \\ 0 & 1 & 0 & \sin\delta\beta & 0 & 0 \\ \sin\delta\beta & 0 & \cos\delta\beta & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\delta\beta & 0 & -\sin\delta\beta \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \sin\delta\beta & 0 & \cos\delta\beta \end{bmatrix}$$
(11)

The variational form of a boundary value problem of the built-up structure is given in Eq. (15). Taking the first variation of both sides of Eq. (15) and noting $\dot{z} = \dot{z}_{V_\Omega}$ + $\dot{z}_{V_\theta}$, Eq. (16) is obtained. Using the fact that $\dot{z} \in Z$ and $a_\Omega(z,\dot{z}) = \ell_\Omega(\dot{z})$, Eq. (16) becomes Eq. (17).

Consider a performance measure in a general form as in Eq. (18). Taking the first variation of Eq. (18), we can obtain Eq. (19). In the direct differential method, Eq. (19) is solved for $\dot{z}$ with the given design velocity fields V_Ω and V_θ. Once the original response z and the first variation $\dot{z}$ are obtained, the configuration design sensitivity expression in Eq. (19) can be evaluated. In the adjoint variable method, an adjoint equation is defined in Eq. (20) and is solved for the adjoint response λ. Since $\dot{z}$ is in the space of kinematically admissible displacements, Eq. (20) can be evaluated at $\bar{\lambda} = \dot{z}$ and Eq. (17) at $\bar{z} = \lambda$, to obtain Eq. (21). Once the design velocity fields are defined, with the original response z and the adjoint response λ, the configuration design sensitivity expression in Eq. (21) can be calculated.

Configuration Design Sensitivity Analysis

$$a_\Omega(z,\bar{z}) = \ell_\Omega(\bar{z}), \qquad \text{for all } \bar{z} \in Z \tag{15}$$

$$[a_\Omega(z,\bar{z})]' = a_\Omega(\dot{z},\bar{z}) + a_\Omega(z,\dot{\bar{z}}) + a'_{V_\Omega}(z,\bar{z}) + a'_{V_\theta}(z,\bar{z})$$

$$= \ell_\Omega(\dot{\bar{z}}) + \ell'_{V_\Omega}(\bar{z}) + \ell'_{V_\theta}(\bar{z}) = [\ell_\Omega(\bar{z})]' \tag{16}$$

$$a_\Omega(\dot{z},\bar{z}) = \ell'_{V_\Omega}(\bar{z}) + \ell'_{V_\theta}(\bar{z}) - a'_{V_\Omega}(z,\bar{z}) - a'_{V_\theta}(z,\bar{z}), \qquad \text{for all } \bar{z} \in Z \tag{17}$$

$$\psi = \int_{\Omega_\tau} g(z_{i\tau}, \nabla z_{i\tau}, z_{i\tau,jk}) \, d\Omega_\tau \tag{18}$$

(A). Direct Differential Method:

$$\psi' = \int_\Omega [g_{z_i}\dot{z}_i + g_{\nabla z_i}\nabla\dot{z}_i + g_{z_{i,jk}}\dot{z}_{i,jk} - g_{z_i}(V_\theta z)_i - g_{\nabla z_i}\nabla(V_\theta z)_i - g_{z_{i,jk}}(V_\theta z)_{i,jk}$$

$$- g_{z_i}(\nabla z_i^T V_\Omega) - g_{\nabla z_i}\nabla(\nabla z_i^T V_\Omega) - g_{z_{i,jk}}(\nabla z_i^T V_\Omega)_{,jk} + \nabla g^T V_\Omega + g(\nabla^T V_\Omega)] \, d\Omega \tag{19}$$

(B). Adjoint Variable Method:

$$a_\Omega(\lambda,\bar{\lambda}) = \int_\Omega [g_{z_i}\bar{\lambda}_i + g_{\nabla z_i}\nabla\bar{\lambda}_i + g_{z_{i,jk}}\bar{\lambda}_{i,jk}] \, d\Omega, \qquad \text{for all } \bar{\lambda} \in Z \tag{20}$$

$$\psi' = \ell'_{u,V_\Omega}(\lambda) + \ell'_{u,V_\theta}(\lambda) - a'_{u,V_\Omega}(z,\lambda) - a'_{u,V_\theta}(z,\lambda) - \int_\Omega [g_{z_i}(V_\theta z)_i + g_{\nabla z_i}\nabla(V_\theta z)_i$$

$$+ g_{z_{i,jk}}(V_\theta z)_{i,jk} - g_{z_i}(\nabla z_i^T V_\Omega) + g_{\nabla z_i}\nabla(\nabla z_i^T V_\Omega) + g_{z_{i,jk}}(\nabla z_i^T V_\Omega)_{,jk}] \, d\Omega$$

$$+ \int_\Omega [\nabla g^T V_\Omega + g(\nabla^T V_\Omega)] \, d\Omega \tag{21}$$

A swept wing model shown in Fig. 4 is considered for the study of configuration design sensitivity analysis. This wing model consists of the truss and membrane design components. The wing is made by aluminum, and is subjected to a uniform pressure (0.556 psi) acting on top of the skin panels. An established finite element code ANSYS is used to create the finite element mesh. Because of the symmetry of the structure and loading, only half of the wing box is analyzed. The finite element model consists of 60 truss elements (STIF 8) and 130 membrane elements (STIF 41).

For a configuration design change, the tip of the swept wing is moved forward as shown in Fig. 4. The design velocity fields are defined so that all ribs (shear panels) that are parallel to the y axis remain parallel while moving. The layout of the spars (shear panels) and the skin panels will then be rotated accordingly. The displacement at the tip of the wing and several stress performance measures are specified. The configuration design sensitivity analysis is carried out using the finite element results obtained from ANSYS. Results presented in Fig 5 show an excellent agreement between the predictions ψ' and actual changes $\Delta\psi$, where $\Delta\psi$ is obtained by the central difference.

Configuration DSA of Swept Wing Model
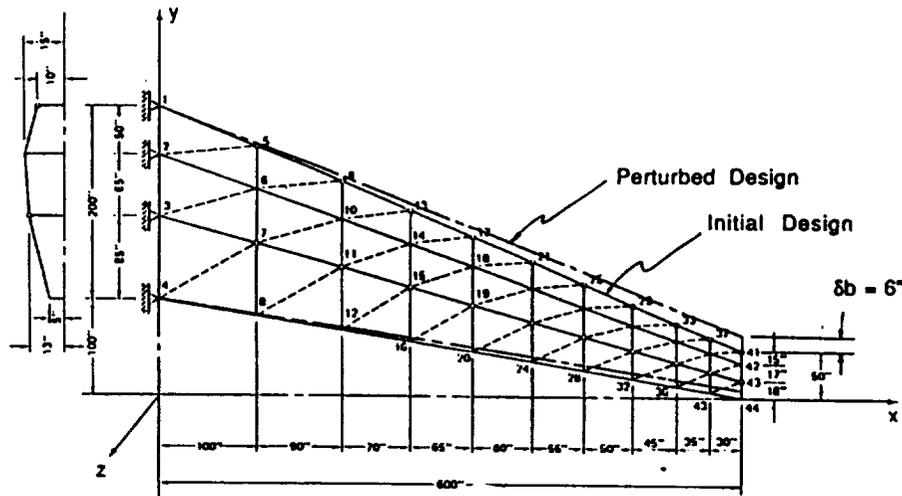
Figure 4 Sw

Performance Measures Specification

No.	Specification	
1	Displacement at node #41 in z direction.	
2	" #42 "	0
3	" #43 "	0
4	" #44 "	0
5	Axial Stress on element #1. (Spar Cap)	0
6	" #2 "	0.0
7	Average Von Mises Stress on element #3 (Skin Panel)	0.1
8	" #4 "	00.1
9	" #5 "	00.0
10	" #6 "	100.0
11	" #7 "	100.0
12	" #8 "	100.0
13	" #9 (Shear Panel)	100.0
14	" #10 "	100.0
15	" #11 "	100.2
16	" #63 (Skin Pane	100.0
17	" #64 "	100.0
18	" #65 "	100.0
19	" #66 "	100.0
20	" #67 "	100.0

Minimize over x the
set $f(x, alpha)$: $h(x,a) =$
$\emptyset$, $g(x, alpha)$ is greater
than or equal to $\emptyset$ where
$f : R(sup n+r)$ approaches
...

Figure 5 Configuration Design Sensitivity ...

48

In summary, a unified configuration design sensitivity analysis of built-up structures has been developed. The configuration design change is identified by the translation, rotation, and shape variation. The material derivative idea of continuum mechanics is used to account for the shape variation. In this paper, a design sensitivity analysis method to treat orientation change of a design component has been developed. The numerical implementation of configuration design sensitivity analysis is carried out by using the results of an established finite element code. The results show that the method leads to an accurate and efficient configuration design sensitivity analysis of the built-up structure.

REFERENCES

1. Topping, B. H. V., "Shape Optimization of Skeletal Structures: A Review," ASCE Journal of the Structural Division, Vol. 109, 1983, pp. 1933-1952.

2. Saka, M. P. and Attili B., "Shape Optimization of Space Trusses," The proceedings of the International Conference on the Design and Construction of Non-conventional Structures (Ed. B.H.V. Topping), Civil-Comp Press, London, 1987, pp. 115-121.

3. Felix, J. and Vanderplaats, G. N., "Configuration Optimization of Trusses Subject to Strength, Displacement, and Frequency Constraints," ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 109, No. 2, June 1987, pp. 233-241.

4. Sandgren, E, Lee, H, and El-Sayed M., "Optimal Design of Mechanical Components with Sizing, Configurational and Topological Consideration," The 1989 ASME Design Technical Conferences - 15th Design Automation Conference, Canada, Sept. 17-21, 1989, Advances in Design Automation, Vol. 2, 1989, pp.101-110.

5. Haug, E. J., Choi, K. K., and Komkov, V., Design Sensitivity Analysis of Structural System, Academic Press, New York, 1986.

6. Choi, K. K. and Santos, J. L. T., "Design Sensitivity Analysis of Nonlinear Structural Systems - I, Theory," International Journal for Numerical Methods in Engineering, Vol. 24, 1987, pp. 2039-2055.

7. Choi, K. K. and Lee, J. H., "Sizing Design Sensitivity Analysis of Dynamic Frequency Response of Vibrating Structures," The 1989 ASME Design Technical Conferences - 15th Design Automation Conference, Canada, Sept. 17-21, 1989, Advances in Design Automation, Vol. 2, 1989, pp. 257-265.

8. Choi, K. K. and Wang S., "Design Sensitivity Analysis of Structural Dynamic Response Using Ritz Sequence," AIAA 31th Structures, Structural Dynamics & Materials Conference, Long Beach, CA, April 2-4, 1990, pp. 385-393.

9. Lee, H. G., Choi, K. K., and Haug, E. J., "Shape Optimal Design of Build-up Structures," Technical Report No. 84-12, Center for Computer Aided Design , The University of Iowa, Iowa City, 1984.

BIFURCATIONS AND SENSITIVITY IN
PARAMETRIC NONLINEAR PROGRAMMING*

Bruce N. Lundberg   and   Aubrey B. Poore

Department of Mathematics   Department of Mathematics

Grand Canyon University   Colorado State University

3300 W. Camelback   Fort Collins, CO 80523

Phoenix, Arizona 85017

1. Introduction. The parametric nonlinear programming problem is that of determining the behavior of solution(s) as a parameter or vector of parameters $\alpha \in \mathcal{R}^r$ varies over a region of interest for the problem

(1.1) $\qquad \underset{x}{\text{Minimize}} \quad \{f(x,\alpha): \ h(x,\alpha) = 0, \ g(x,\alpha) \geq 0\},$

where $f: \mathcal{R}^{n+r} \to \mathcal{R}$, $h: \mathcal{R}^{n+r} \to \mathcal{R}^q$ and $g: \mathcal{R}^{n+r} \to \mathcal{R}^p$ are assumed to be at least twice continuously differentiable. Some of these parameters may be fixed but not known precisely and others may be varied to enhance the performance of the system. In both cases a fundamentally important problem in the investigation of *global sensitivity* of the system is to determine the stability boundaries of the regions in parameter space which define regions of qualitatively similar solutions. The objective in this work is to explain how numerical continuation and bifurcation techniques can be used to investigate the parametric nonlinear programming problem in a global sense. Thus we first convert the problem (1.1) to a closed system of parameterized nonlinear equations whose solution set contains all local minimizers of the original problem. This system, which will be represented as $F(z,\alpha) = 0$, will include all Karush-Kuhn-Tucker and Fritz John points, both feasible and infeasible solutions, and relative minima, maxima, and saddle points of (1.1). The local existence and uniqueness of a solution path $(z(\alpha), \alpha)$ of this system as well as the solution type persist as long as a singularity in the Jacobian $D_z F(z,\alpha)$ is not encountered. Thus we first characterize the nonsingularity of this Jacobian in terms of conditions on the problem (1.1) itself. We then describe a class of efficient predictor-corrector continuation procedures for tracing solution paths of the system $F(z,\alpha) = 0$ which are tailored specifically to the parametric programming problem. Finally, these procedures and the obtained information will be illustrated within the context of design optimization.

2. Systems Formulation and Bifurcation Problems. If $M = diag(\mu_1, \ldots, \mu_p)$ is a diagonal matrix and $\mathcal{L} = \mu_{p+1} f(x,\alpha) - \sum_{j=1}^{q} \lambda_j h_j(x,\alpha) - \sum_{i=1}^{p} \mu_i g_i(x,\alpha)$ is the Lagrangian, then any solution of the Fritz John or Karush-Kuhn-Tucker first order necessary conditions is a solution of the closed system [8]

(2.1) $\qquad F(z,\alpha) = \begin{bmatrix} \nabla_x \mathcal{L}(z,\alpha) \\ -h(x,\alpha) \\ -Mg(x,\alpha) \\ \mu^T \mu + \lambda^T \lambda - \beta_0^2 \end{bmatrix} = 0, \ \text{where } z = \begin{bmatrix} x \\ \lambda \\ \mu \end{bmatrix},$

and β_0 is a fixed constant. In the presence of a smooth F, a necessary condition for the existence of multiple solution branches to the system $F(z, \alpha) = 0$ in each neighborhood of a solution (z_0, α_0) is that the Jacobian $D_z F(z_0, \alpha_0)$ be singular. Since solution type (minimum, maximum, saddle point; feasible or infeasible point) can change only at such a singularity, we now give necessary and sufficient conditions for $D_z F$ to be nonsingular.

Theorem 2.1. *[8] Let (z_0, α_0) be a solution of $F(z, \alpha) = 0$, defined by 2.1, where f, g and h are C^k, $(k \geq 2)$, in a neighborhood of (z_0, α_0). Define two index sets $\bar{A}$ and A and a corresponding tangent space $\bar{T}$ by*

$$\bar{A} = \{i : 1 \leq i \leq p, g_i(z_0, \alpha_0) = 0\}, \qquad A = \{i \in \bar{A} : \mu_i^0 \neq 0\}$$

$$\bar{T} = \{y \in \mathcal{R}^n : D_x h(z_0, \alpha_0)y = 0, \ D_x g_i(z_0, \alpha_0)y = 0 \ (i \in \bar{A})\}$$

Then a necessary and sufficient condition that $D_z F(z_0, \alpha_0)$ be nonsingular is that each of the following three conditions hold:

a) *$\bar{A} = A$;*

b) *$S := \{\nabla_x g_i(z_0, \alpha_0)\}_{i \in A} \cup \{\nabla_x h_j(z_0, \alpha_0)\}_{j=1}^q$ is a linearly independent collection of $q + |\bar{A}|$ vectors where $|\bar{A}|$ denotes the cardinality of $\bar{A}$;*

c) *The Hessian of the Lagrangian $\nabla_x^2 \mathcal{L}$ is nonsingular on the tangent space $\bar{T}$ at (z_0, α_0).*

Having stated this theorem, several comments are in order. If z_0 is a Fritz John or Karush-Kuhn-Tucker point, then condition (a) is called strict complementarity ($g_i(z_0, \alpha_0) = 0$ implies μ_i^0 is positive) and condition (b) is the linear independence constraint qualification. Furthermore, if in addition to conditions (a) and (b), the Hessian of the Lagrangian is positive definite on the tangent space $\bar{T}$, then z_0 is a local minimizer at α_0.

An equivalent and more computationally efficient method for tracing solution path segments of (2.1) along which $D_z F$ is nonsingular is to use an "active set" strategy in which a purely equality constrained problem is considered, with active inequality constraints playing the role of additional equality constraints. This amounts to deleting inactive inequality constraints and corresponding (zero) multipliers from the definitions of $\mathcal{L}$ and F and replacing the components $-\mu_i g_i$ of F by $-g_i$ for each $i \in \bar{A}$. In case a multiplier for an active inequality constraint changes sign along a path segment one has passed a singular point on the path due to a violation of a) in Theorem 2.1. Paths branching from such a point correspond to various choices of the active set.

3. Numerical Continuation and Bifurcation Methods. Since the subject of numerical continuation and bifurcation methods has a formidable literature and since excellent introductions to this subject area can be found in the books of Allgower and Georg [2], Keller [5] and Rheinboldt [9], our objective in this section is to briefly introduce these techniques and show how they can be tailored to the parametric programming problem.

To describe the predictor-corrector continuation methods, let $w = (z, \alpha)$ so that the problem is that of tracing solution paths of an underdetermined system of nonlinear equations $F(w) = 0$ where $F : \mathcal{R}^{m+1} \rightarrow$

$\mathcal{R}^m$. Assume that $F(w) = 0$ is continuously differentiable, has a smooth solution path $P = \{w \in \mathcal{R}^{m+1} : w = \Psi(s), s \in I\}$ where I is an interval of real numbers, and that the path is nonsingular in that $[D_w F|_{w \in P}]$ is of full rank. Most path following algorithms generate a sequence $\{(w_k, s_k)\}_{k=0}^N$ where w_k is a point on or near the path and w_0 is a known solution of $F(w) = 0$. To go from a point w_k to a point w_{k+1}, one uses current and previous information to obtain a predicted point, say wp_{k+1}, which becomes the starting point for a Newton-like correction iteration which terminates with a solution w_{k+1}.

Given a point w_k on the path, one predicts a new point by using a predictor of the form $wp_{k+1} = w_k + \Delta s d(\Delta s)$. The prediction direction d is typically chosen to be an oriented unit tangent T_k, which is a solution to $[D_w F(w_k)]T_k = 0$. However, a more robust and efficient prediction strategy which uses current and previous tangents has been developed by the authors [6] and will be used below. Once a predicted point is obtained, the correction back to the path can be based on a Newton-like solution of the augmented system

$$(3.1) \qquad G(w) = \begin{bmatrix} F(w) \\ N(w) = (w - wp_{k+1})^T d(\Delta s) \end{bmatrix} = 0$$

which confines the correction to a hyperplane orthogonal to the prediction direction $d(\Delta s)$.

It follows from this brief description that the two main computational problems in a predictor-corrector step are the determination of Newton corrections Δw for the system (3.1) and the computation of the tangent vector T_{k+1}. (The computation of the tangent is essentially free after one computes the Newton correction.) We now describe how to compute these in a way specifically tailored to the parametric programming problem. In light of the comments at the end of section 2 it suffices to consider the case of an equality constrained problem. For this case the techniques presented below reduce the matrix algebra in a continuation step to that of the Lagrangian matrix $W = \begin{bmatrix} \nabla_x^2 \mathcal{L} & -(D_x h)^T \\ -D_x h & 0 \end{bmatrix}$ associated with the nonlinear programming problem.

To solve the system $[D_w G(w)]\Delta w = -G(w)$ for a Newton correction Δw, we give a variant of the bordering algorithm of Keller [5], which also accounts for the presence of the additional augmenting equation $B(\lambda, \mu_1) = \lambda^T \lambda + \mu_1^2 - \beta_0^2 = 0$ in (2.1) and (3.1). Let $\tilde{y}, \tilde{v}, \tilde{u} \in \mathcal{R}^{n+t}$ be solutions of

$$(3.2) \qquad W\tilde{y} = -\begin{bmatrix} \nabla_x \mathcal{L}(x, \alpha) \\ -h(x, \alpha) \end{bmatrix}, \qquad W\tilde{v} = -\begin{bmatrix} \nabla_x f(x, \alpha) \\ 0 \end{bmatrix}, \qquad \text{and} \qquad W\tilde{u} = -\frac{\partial}{\partial \alpha} \begin{bmatrix} \nabla_x \mathcal{L}(x, \alpha) \\ -h(x, \alpha) \end{bmatrix},$$

respectively. Furthermore, let ℓ, y, v and u be vectors in $\mathcal{R}^{n+t+2}$ defined by $\ell = \begin{pmatrix} 0 \\ \lambda \\ \mu_1 \\ 0 \end{pmatrix}, y = \begin{pmatrix} \tilde{y} \\ 0 \\ 0 \end{pmatrix}$, $v = \begin{pmatrix} \tilde{v} \\ 1 \\ 0 \end{pmatrix}$, and $u = \begin{pmatrix} \tilde{u} \\ 0 \\ 1 \end{pmatrix}$. Then it can be shown [7] that a Newton correction step for (3.1) is given by

$$(3.3) \qquad \Delta w = y + sv + tu, \qquad \text{where} \qquad \begin{bmatrix} \ell^T v & \ell^T u \\ d^T v & d^T u \end{bmatrix} \begin{bmatrix} s \\ t \end{bmatrix} = -\begin{bmatrix} \frac{1}{2}B + \ell^T y \\ N + d^T y \end{bmatrix}.$$

If the vectors ℓ, y, v and u are computed at w_{k+1} (or some approximation to w_{k+1}) the tangent at w_{k+1} can be computed via

$$(3.4) \qquad T_{k+1} = \pm \left[(\ell^T v)u - (\ell^T u)v \right] / \|(\ell^T v)u - (\ell^T u)v\|_2.$$

The sign depends on orientation and is changed when a fold point is encountered along the path.

The nonsingularity of W is also characterized by conditions a), b) and c) given in Theorem 2.1; the nonsingularity of $D_w G$ and of the 2×2 matrix in (3.3) follow from the nonsingularity of W as long as the prediction direction d is not orthogonal to the null space of $D_w F$. The latter is the case for our predictor as long as Δs is not too large. Fold points or bifurcation points along the path are indicated by a singularity in W, and may be detected, distinguished and handled by methods discussed by Keller [5] together with methods we now describe.

The methods for solving the linear systems in (3.2) can be based on various linear algebra techniques in nonlinear programming, modified to account for the possibility that $\nabla_x^2 \mathcal{L}_T$ maybe indefinite on some segments of the path. One must also adapt these linear algebra techniques to determine sign changes in three important sets which determine critical point type at regular points: (A) sign $g_i(x, \alpha)$ for $i \in \{1, \ldots, p\}$, (B) sign μ_i for $i \in \{1, \ldots, p+1\}$ and (C) the signs of the eigenvalues of $\nabla_x^2 \mathcal{L}_T$, the restriction of the Hessian of the Lagrangian to the tangent space of the active constraints. To see how one may monitor and detect changes in the signature of $\nabla_x^2 \mathcal{L}_T$ in the course of using a generalized null space method for solving (3.2), assume conditions a), b) and c) of Theorem 2.1 hold. Then the $k \times n$ matrix $A^T = \left\{ \begin{array}{l} D_x h \\ D_x g_i \quad i \in \bar{A} \end{array} \right\}$ is of full rank. In a null space method one first computes matrices $Y \in \mathcal{R}^{n \times k}$ and $Z \in \mathcal{R}^{n \times (n-k)}$ of full rank such that [Y:Z] is nonsingular, $A^T y = I$ and $A^T Z = 0$. In the course of solving (3.2) by such a method one must form and factor the $(n-k) \times (n-k)$ matrix $Z^T \nabla_x^2 \mathcal{L} Z$, whose signature is the same as that of $\nabla_x^2 \mathcal{L}_T$. Given that $Z^T \nabla_x^2 \mathcal{L} Z$ may be indefinite in the continuation process, one generally would compute the LDL^T factorization of this matrix by using, e.g., the Bunch-Kaufman algorithm [3, § 4.4]. Here, the matrix D is a block diagonal matrix with 1×1 and 2×2 blocks whose signature is easily computed and is the same as that of $Z^T \nabla_x^2 \mathcal{L} Z$.

Once a change in a sign in (A-C) is detected, a singularity is detected which one must deal with appropriately, reversing the orientation in the case of a fold point or switching branches at a bifurcation point. One particularly easy aspect of this problem is the case associated with a loss of strict complementarity: to switch branches, one simply activates or de-activates a constraint. Further analysis and classification of these fold and bifurcation points as well as methods for detecting them numerically can be found in [8,10] and in a forthcomming paper [7].

4. A Numerical Example. As a simple illustration of the above procedures we consider the following problem from design optimization [4]

(4.1) $\qquad \underset{(d, h)}{\text{Minimize}} \quad \{d: \quad \nabla_d E(d, h; p) = 0, \qquad h \leq 1.5, \quad h \geq 0\}$

where $E(d, h; p) = -pd + \left(\sqrt{1 + h^2} - \sqrt{1 + (h - d)^2}\right)^2 / \sqrt{1 + h^2}$, and p is a parameter. This problem is used to model the determination of the unloaded height h of a simple two bar planar truss with semi-span

53

1 which minimizes the displacement d under a fixed load p. The solution paths $z(p)$ of (2.1) were tracked using our continuation method. The following plot gives the displacement d as p varies.

Fig 4.1: Feasible Solutions of (2.1)
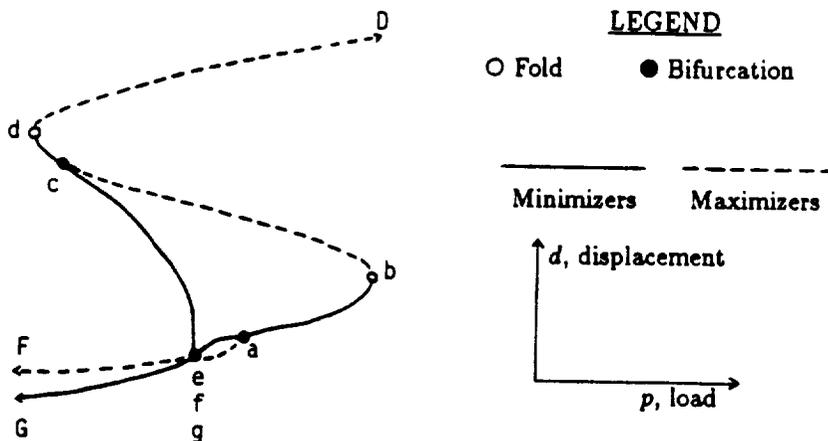
This plot represents a projection of the feasible solutions of (2.1) into the (p, d) plane, and the dot labeled with e, f and g indicates three distinct bifurcation points with $p = d = 0$ and $h = 0$, 1.41 and 1.5 respectively. Bifurcation points a, c, e and g result from a loss of strict complementarily in which an inequality constraint becomes weakly active. The path of maximizers branching from point a corresponds to $h \leq 1.5$ active and $\mu_1 < 0$, and changes type at the singular point g, becoming the path of minimizers labeled G. The other path branching from point a passes through f, across which the one eigenvalue of $\nabla_x^2 \mathcal{L}_f$ changes sign. At f there is a change in type resulting in the path of maximizers labeled F.

Extreme sensitivity of the solution of (1.1) to variations in p occurs at the fold points b and d ($p = \pm .3704$) at which there is a loss of linear independence in the active constaint gradients, and $\mu_2 = 0$. This is also the case at the bifurcation point e, where in addition, strict complementarily is violated. One cannot compute near or past these points without the normalization $B(\lambda, \mu) = \lambda^T \lambda + \mu^T \mu - \beta_0^2 = 0$, since near these points an unnormalized multiplier is unbounded. When the system is at a state near these points, small variations in load p can result in very large changes in the solution, or the loss of (local) existence of a solution. The latter case is illustrated near b where increasing the parameter past $p = .3704$ results in the loss of the solution and a "snap through" of the truss to a state represented by path D. Similar behavior occurs near d and e.

Not pictured above are branches of infeasible solutions of (2.1) emerging at a, c, e and g ($h > 1.5$ or $h < 0$). (In some problems such paths may provide the opportunity for further branching to other feasible paths.) A path of feasible singular points with $p = d = 0$ branches from e through f to g, and can be parameterized by μ_2. The solutions to (4.1) need only be stationary points of the potential energy $E(d, h; p)$. However, all path segments, exclusive of the segments from d to e and from c to b, do correspond to physical states of the system (where E is minimized).

In this paper we have described how continuation techniques can be tailored to the parametric nonlinear programming problem and used to investigate the global dependence of the solution on a parameter. The use of the system (2.1), including the normalization $B(\lambda, \mu) = 0$ enables the computation to proceed through singular points where solution type may change and branching and/or sensitivity of solutions of (1.1) occurs. Computing paths of maximizers, saddle points and singular points enables one to locate regions of sensitivity, multiple operating states, and disconnected paths of minimizers.

We think that these methods have great potential to facilitate parametric study in nonlinear programming and, for example, in nonlinear optimal control. Further testing and development of the numerical methods on some tractable model problems is needed, as well as further analysis of the behavior which can be expected at more complex singularities.

REFERENCES

[1] Allgower, E. L. and K. Georg, *Introduction to Numerical Continuation Methods*, Springer-Verlag, to appear.

[2] Fletcher, R., *Practical Methods of Optimization*, Second Edition, John Wiley and Sons, New York, 1987.

[3] Golub, G. H. and C. G. Van Loan, *Matrix Computations*, Second Edition, The Johns Hopkins University Press, Baltimore, 1989.

[4] Rao, J. R. J. and P. Y. Papalambros, Extremal Behavior of One Parameter Families of Optimal Design Models, ASME Design Automation Conference, Montreal, Sept 17-20, 1989.

[5] Keller, H. B., *Numerical Methods in Bifurcation Problems*, Springer-Verlag, Berlin, 1987.

[6] Lundberg, B. N. and A. B. Poore, Variable order Adams-Bashforth predictors with an error-stepsize control for continuation methods, to appear in SIAM J. and Sci. and Stat. Comp., 1991.

[7] Lundberg, B. N. and A. B. Poore, Numerical continuation and bifurcation methods for the parametric nonlinear programming problem, in preparation.

[8] Poore, A. B. and C. A. Tiahrt, Bifurcation problems in nonlinear parametric programming, *Mathematical Programming*, 39 (1987), 189-205.

[9] Rheinboldt, W. C., *Numerical Analysis of Parameterized Nonlinear Equations*, John Wiley, New York, 1985.

[10] Tiahrt, C. A. and A. B., Poore, A bifurcation analysis of the nonlinear parametric programming problem, to appear in *Mathematical Programming*, 1990.

IMPROVED CALCULATION OF OPTIMUM DESIGN SENSITIVITY*

Abdon E. Sepulveda and *Lucien A. Schmit***
University of California, Los Angeles, California
4531 Boelter Hall, UCLA, Los Angeles, CA 90024-1593

Introduction

Optimum design parameter sensitivity analysis has become an important topic in recent years. The principal reasons for obtaining optimum design sensitivity information with respect to various problem parameters are (1) to predict revised optimum designs, associated with specified perturbations of the problem parameters, without re-optimizing the problem and, (2) to provide sensitivity information in multilevel optimization strategies. Methods for calculating these derivatives have been proposed by several authors (see Refs. 1-5). The most important drawback in estimating parameter sensitivities by the current methods is that they do not allow for changes in the active constraint set. Changes in the active constraint set produce discontinuities in the optimum design sensitivities, and therefore, a correct formulation has to be cast in terms of directional derivatives (Refs. 6-7). This paper addresses optimum design parameter sensitivity analysis in terms of directional derivatives so as to include possible discontinuities and it also presents a method for estimating optimum design sensitivities using finite differences.

Problem Formulation

The general optimization problem considered in this work has the following form

$$\underset{Y}{\text{Min}} \quad f(Y,P)$$

$$\text{s.t.} \quad g_j(Y,P) \geq 0 \quad\quad j = 1,...,m \tag{1}$$

where $Y = (Y_1,...,Y_n)$ is the vector of design variables and $P = (P_1,...,P_p)$ is the vector of design parameters.

It is assumed that for a fixed set of parameters P, the optimization problem has been solved, so that the vector of optimum design variables $Y^*(P)$ is known, and that this optimal solution satisfies the first order Kuhn Tucker conditions given by

$$\nabla_Y f(Y^*(P),P) - \sum_{j=1}^{m} \lambda_j(P) \nabla_Y g_j(Y^*(P),P) = 0 \tag{2a}$$

$$\lambda_j(P) g_j(Y^*(P),P) = 0 \quad\quad j = 1,...,m \tag{2b}$$

$$g_j(Y^*(P),P) \geq 0 \quad\quad j = 1,...,m \tag{2c}$$

$$\lambda_j(P) \geq 0 \quad\quad j = 1,...,m \tag{2d}$$

---

* Postdoctoral Research Associate.

** Professor of Engineering and Applied Science.

where $\lambda(P)$ denotes the vector of Lagrange multipliers at the optimum design $Y^*(P)$ for the given vector of parameters P.

If the vector P is perturbed, the set of active constraints for the new optimum will change depending on the direction of move for P. This change induces discontinuities for the sensitivities of the optimal solution, or in mathematical terms, the derivatives $\frac{dY(P)}{dP}$ are not unique. References 6 and 7 prove that even if the derivatives are not unique, the directional derivatives exist. In what follows the optimal sensitivity problem is formulated in terms of directional derivatives, defined as

$$Y'_{\delta P}(P) = \lim_{t \to 0^+} \frac{Y(P + t\,\delta P) - Y(P)}{t} \qquad (3)$$

where δP represents a unit vector ($|\,\delta P\,| = 1$) in a prescribed direction of change for the vector of parameters P.

Sensitivity Using Kuhn Tucker Conditions

If the vector of parameters P is perturbed by $t\delta P$, $t > 0$, the Kuhn Tucker conditions (Eq. (2)) must remain satisfied. Moreover, since the sensitivities are obtained when $t \to 0$ only first order approximations for Y and λ in terms of t are required, and only linear terms in t should be retained.

Using the approximations

$$Y(P + t\,\delta P) = Y(P) + t\,Y'_{\delta P}(P) \qquad (4a)$$

$$\lambda(P + t\,\delta P) = \lambda(P) + t\,\lambda'_{\delta P}(P) \qquad (4b)$$

Eq. (2) for the perturbed problem becomes

$$\nabla_Y f(Y(P) + tY'_{\delta P}(P), P + t\,\delta P) - \sum_{j=1}^{m} (\lambda_j(P) + t\,\lambda'_{j\delta P}(P))(\nabla_Y g_j(Y(P) + t\,Y'_{\delta P}(P), P + t\,\delta P) = 0 \qquad (5a)$$

$$(\lambda_j(P) + t\,\lambda'_{j\delta P}(P))(g_j(Y(P) + t\,Y'_{\delta P}(P), P + t\,\delta P)) = 0 \qquad j = 1, \ldots, m \qquad (5b)$$

$$g_j(Y(P) + t\,Y'_{\delta P}(P), P + t\,\delta P) \geq 0 \qquad j = 1, \ldots, m \qquad (5c)$$

$$\lambda_j(P) + t\,\lambda'_{j\delta P}(P) \geq 0 \qquad j = 1, \ldots, m \qquad (5d)$$

Expanding g_j in first order Taylor series and retaining only linear terms in t, Eqs. (5b)-(5c) become:

$$\lambda_j(P)\,[g_j(Y(P), P) + t\,(\nabla_Y g_j(Y(P), P)\,Y'_{\delta P}(P) + \nabla_P g_j(Y(P), P)\delta P] + t\,\lambda'_{\delta P}(P)\,g_j(Y(P), P) = 0 \qquad (6a)$$

$$g_j(Y(P), P) + t\,(\nabla_Y g_j(Y(P), P)\,Y'_{\delta P}(P) + \nabla_P g_j(Y(P), P)\,\delta P) \geq 0 \qquad (6b)$$

$$\lambda_j(P) + t\,\lambda'_{j\delta P}(P) \geq 0 \qquad (6c)$$

Several special cases arise from these previous equations when $t \to 0^+$:

(1) $\lambda_j(P) = 0$, $\quad g_j(Y(P), P) = 0$ $\qquad$ (active constraint, degenerate case)

$$\nabla_Y g_j(Y(P), P) Y'_{\delta P}(P) + \nabla_P g_j(Y(P), P) \delta P \geq 0 \qquad (7a)$$

$$\lambda'_{j\delta P}(P) \geq 0$$

(2) $\lambda_j(P) > 0$, $\quad g_j(Y(P), P) = 0$ $\qquad$ (active constraint, non degenerate case)

$$\nabla_Y g_j(Y(P), P) Y'_{\delta P}(P) + \nabla_P g_j(Y(P), P) \delta P = 0 \qquad (7b)$$

$$\lambda'_{j\delta P}(P) \qquad \text{has no sign constraint}$$

(3) $\lambda_j(P) = 0$, $\quad g_j(Y(P), P) > 0$ $\qquad$ (not active constraints)

$$\nabla_Y g_j(Y(P), P) Y'_{\delta P}(P) + \nabla_P g_j(Y(P), P) \delta P \qquad \text{has no sign constraint} \qquad (7c)$$

$$\lambda'_{j\delta P}(P) \qquad \text{has no sign constraint}$$

Equations (7) show that only degenerate active constraints are allowed to leave the set of active constraints, and that non active constraints can only become degenerate active constraints. Therefore, it is not necessary to include passive constraints in the sensitivity analysis. Since the Kuhn Tucker conditions are satisfied for the unperturbed problem (Eq. (2a)), the expansion of Eq. (5a) in first order Taylor series, leads to

$$\nabla^2_{YY} L(Y(P), P) Y'_{\delta P}(P) + \nabla^2_{YP} L(Y(P), P) \delta P - \sum_{j=1}^{m} \lambda'_{j\delta P}(P) \nabla_Y g_j(Y(P), P) = 0 \qquad (8)$$

where

$$L(Y(P), P) = f(Y(P), P) - \sum_{j=1}^{m} \lambda_j(P) g_j(Y(P), P) \qquad (9)$$

is the Lagrangian function and $\nabla^2_{YY} L = \left[\frac{\partial L}{\partial Y_i \partial Y_j} \right]$ and $\nabla^2_{YP} L = \left[\frac{\partial L}{\partial Y_i \partial P_j} \right]$.

Equations (7) and (8) are the conditions that the directional derivatives must satisfy for the perturbed problem. Since only the active constraints are to be considered, these equations lead to the following conditions which must be satisfied by $Y'_{\delta P}(P)$ and $\lambda'_{\delta P}(P)$:

$$\nabla^2_{YY} L(Y(P), P) Y'_{\delta P}(P) + \nabla^2_{YP} L(Y(P), P) \delta P - \sum_{j \in J_1} \lambda'_{j\delta P}(P) \nabla_Y g_j(Y, P), P)$$

$$- \sum_{j \in J_2} \lambda'_{j\delta P}(P \nabla_Y g_j(Y(P), P) = 0 \qquad (10a)$$

$$\nabla_Y g_j(Y(P), P) Y'_{\delta P}(P) + \nabla_P g_j(Y(P), P) \delta P = 0 \quad j \in J_1 \qquad (10b)$$

$$\nabla_Y g_j(Y(P), P) Y'_{\delta P}(P) + \nabla_P g_j(Y(P), P) \delta P \geq 0 \quad j \in J_2 \qquad (10c)$$

$$\lambda'_{j\delta P}(P) \geq 0 \quad j \in J_2 \qquad (10d)$$

where the sets of indices J_1 and J_2 are given by

$$J_1 = \{ j \mid \lambda_j (P) > 0, \quad g_j (Y(P), P) = 0 \}$$

$$J_2 = \{ j \mid \lambda_j (P) = 0, \quad g_j (Y(P), P) = 0 \}$$

It is easily seen that if all active constraints are non-degenerate ($J_2 = \phi$; i.e. the set J_2 is empty), then the set of active constraints remains invariant and no discontinuities exist. Therefore, if the strict complementary condition holds at the optimum, the Kuhn Tucker based methods presented in Refs. 1 and 2 will give the correct optimum design sensitivities.

The solution of Eq. (10) for $Y'_{\delta P} (P)$ and $\lambda'_{\delta P} (P)$ is easily obtained if it is observed that these equations (Ref. 7) correspond to the Kuhn Tucker conditions for an optimization problem where the objective function is the quadratic approximation of the Lagrangian and the constraints are the linearized active constraints. Thus, the set of conditions given by Eq. (10) is equivalent to the solution of the problem

$$\operatorname*{Min}_{Z} \ \frac{1}{2} Z^T \nabla_{YY}^2 L \ (Y(P), P) Z \ + \ Z^T \nabla_{YP}^2 L (Y(P), P) \, \delta P$$

$$\text{s.t.} \quad \nabla_Y g_j (Y(P), P) \, Z \ + \ \nabla_P g_j (Y(P), P) \, \delta P = 0 \quad j \in J_1 \tag{11}$$

$$\nabla_Y g_j (Y(P), P) \, Z \ + \ \nabla_P g_j (Y(P), P) \, \delta P \geq 0 \quad j \in J_2$$

If the optimal solution for this problem is denoted by Z^* with the associated Lagrange multipliers represented by γ^*, then

$$Y'_{\delta P} (P) = Z^* \tag{12a}$$

$$\lambda'_{\delta P} (P) = \gamma^* \tag{12b}$$

To further illustrate the concepts revealed by the foregoing analysis, consider the following one design variable, single parameter example from Ref. 4.

$$\operatorname*{Min}_{Y} \ f(Y, P) = 2Y^2 - 2YP + P^2 + 4Y - 4P$$

$$\text{s.t.} \quad g(Y, P) = Y + 4P \geq 0$$

Figure 1 shows the optimal solution $Y^*(P)$, the optimal functions $f(Y^*, P)$ and $g(Y^*, P)$ and the Lagrange multiplier $\lambda^* (P)$ as a function of the parameter P. For $P < \frac{2}{9}$, the constraint g is active and the Lagrange multiplier λ is positive, but for $P > \frac{2}{9}$, the constraint becomes passive and $\lambda = 0$. At $P = \frac{2}{9}$ the strict complementary condition does not hold and therefore the derivatives are not unique. At this point only directional derivatives, which in this case correspond to the left and right derivatives, exist. For this simple example problem (at $P = \frac{2}{9}$) the quadratic problem given by Eq. (11) leads to

Case $\delta P = 1$ (right derivative)

Min $2Z^2 - 2Z$

s.t. $Z + 4 \geq 0 \quad \Rightarrow \quad Y'_{\delta P}(2/9) = \frac{1}{2}, \quad \lambda'_{\delta P}(2/9) = 0$

Case $\delta P = -1$ (left derivative)

Min $2Z^2 + 2Z$

s.t. $Z - 4 \geq 0 \quad \Rightarrow \quad Y'_{\delta P}(2/9) = 4, \quad \lambda'_{\delta P}(2/9) = 18$

When numerical techniques are used to determine the optimal solution $Y^*(P)$ and the corresponding Lagrange multipliers $\lambda(P)$ for problem (1), the condition $\lambda_j = 0$, for an active constraint, requires further analysis. The Kuhn Tucker conditions (Eq. (2a)), show that the gradient of the objective function can be written as a non-negative linear combination of the gradients of the active constraint at the optimum. Assuming that the gradients of the active constraints are linearly independent (normality), then, if one of the Lagrange multipliers is zero, this implies that the associated constraint gradient does not contribute to the non-negative linear combination of gradients in terms of which ∇f is expressed. Therefore, the set of indices J_2 should include all constraints such that $\dfrac{\lambda_j}{\sum\limits_i \lambda_i}$ is less than or equal to a prescribed threshold.

Finite Differences

An alternative procedure for obtaining the optimal design sensitivities, (Ref. 4) is to use finite differences for small perturbations of the parameter P. Since only directional derivatives are sought, the direction δP is prescribed and therefore, perturbations for t rather than for P must be considered. The first order Kuhn Tucker optimality conditions for the perturbed problem are given by

$$\nabla_Y f(Y(P + t\delta P), P + t\,\delta P) - \sum_{j=1}^{m} \lambda_j(P + t\,\delta P)\,\nabla_Y g_j(Y(P + t\,\delta P), P + t\,\delta P) = 0 \tag{13a}$$

$$\lambda_j(P + t\,\delta P)\,g_j(Y(P + t\,\delta P), P + t\,\delta P) = 0 \quad j = 1,...,m \tag{13b}$$

$$g_j(Y(P + t\,\delta P), P + t\,\delta P) \geq 0 \quad j = 1,...,m \tag{13c}$$

$$\lambda_j(P + t\,\delta P) \geq 0 \quad j = 1,...,m \tag{13d}$$

In a finite difference scheme, the new optimum $Y(P + t\,\delta P)$ and the associated Lagrange multipliers $\lambda(P + t\,\delta P)$ are to be found such that the conditions imposed by Eqs. (13) are preserved. If the original functions f and $g_j, j = 1,...,m$ are replaced by approximate explicit functions $\bar{f}$ and $\bar{g}_j$ in terms of Y, $j = 1,...,m$, then Eqs. (13) correspond to the Kuhn Tucker conditions of an approximate problem of the form

$$\underset{z}{\text{Min}} \quad \bar{f}(Z, P + t\,\delta P)$$

$$\text{s.t.} \quad \bar{g}_j(Z, P + t\,\delta P) \geq 0 \quad j = 1,...,m \tag{14}$$

If the optimal solution for this problem is denoted by Z^* and the associated Lagrange multipliers are represented by γ^*, then

$$Y'_{\delta P}(P) \approx \frac{Z^* - Y^*(P)}{t} \tag{15a}$$

$$\lambda^*_{\delta P}(P) \approx \frac{\gamma^* - \lambda(P)}{t} \tag{15b}$$

Examination of Eq. (13a) shows that the approximate functions $\bar{f}$ and $\bar{g}_j$, $j = 1, \ldots, m$, should be accurate not only for the function values but for gradients as well. Also, from the formulation of the approximate problem given by Eq. (14), it is clearly seen that these approximations are needed only with respect to the variables Y and not with respect to P.

Quadratic Approximations

A natural choice for approximating the function and its first derivatives, is to use a second order Taylor series expansion with respect to Y. For a generic function, h, this approximation about $Y^*(P)$ has the form:

$$\bar{h}\,(Z, P + t\,\delta P) = h\,(Y(P), P + t\,\delta P) + \nabla_Y h\,(Y(P), P + t\,\delta P)\,(Z - Y(P))$$

$$+\frac{1}{2}\,(Z - Y(P))^T\,\nabla^2_{YY}\,h(Y(P), P + t\,\delta P)\,(Z - Y(P)) \tag{16}$$

and the approximate problem becomes

$$\underset{z}{\text{Min}} \quad \nabla_Y f(Y(P), P + t\,\delta P) +\frac{1}{2}\,(Z - Y(P))^T\,\nabla^2_{YY} f(Y(P), P + t\,\delta P)\,(Z - Y(P))$$

$$\text{s.t.} \quad g_j\,(Y(P), P + t\,\delta P) + \nabla_Y g_j\,(Y(P), P + t\,\delta P)\,(Z - Y(P)) \tag{17}$$

$$+\frac{1}{2}\,(Z - Y(P))^T\,\nabla^2_{YY}\, g_j\,(Y(P), P + t\,\delta P)\,(Z - Y(P)) \geq 0 \qquad j = 1, \ldots, m$$

which corresponds to the second order approximation of the original problem about $Y^*(P)$, at the optimum. This is similar to the second order method given in Ref. 4.

Numerical Results

The methods presented in this paper are demonstrated for the 10-bar truss structure shown in Fig. 2. The cross-sectional areas of the members are the design variables and the objective function to be minimized in the total weight. Constraints are imposed on the stress of each member. The allowable stress is 25,000 lb/in^2 for all members with the exception of member 9. Side constraints are such that 0.1 in$^2 \leq A_i \leq 25$in^2, $i = 1, \ldots, 10$.

Figure 3 shows the response ratios for the stress constraints (on members 2, 9 and 10) as a function of the allowable stress in member 9 ($\overline{\sigma}_9$). It is observed that when $\overline{\sigma}_9 < 30,000$ the stress constraint in member 10 is passive while for $\overline{\sigma}_9 > 30,000$, this constraint becomes active. The opposite effect is observed for the lower bound side constraint on member 10 (from active to passive). The change in the set of active constraints produces discontinuities for the optimum design sensitivities at $\overline{\sigma}_9 = 30,000$ lb/in^2.

Tables 1 and 2 show the right and left derivatives for the optimum design with respect to $\overline{\sigma}_9$. The first column correspond to the exact sensitivities using the Kuhn Tucker conditions. The second and third columns correspond to the approximate derivatives computed via finite differences using a second order approximation. For the third column only diagonal second order terms were retained. For both cases the step was set to 500.

As expected, the sensitivities are not continuous, and the Kuhn Tucker formulation gives the correct directional derivatives. The finite difference results show a strong agreement with respect to the exact solution, even when only second order diagonal terms are retained.

Conclusions

A general procedure for calculating the sensitivity of a optimized design to various problem parameter has been presented. The method is based on the first order Kuhn Tucker conditions and possible discontinuities in the derivatives are taken into account. A finite difference approach is also presented based on second order information. The quadratic method proved to be very accurate in estimating the optimum sensitivities. This method is particularly attractive since no distinction has to be made between degenerate and non degenerate active constraints.

References

1. Armacost, R.L., Fiacco, A.V., "Computational Experience in Sensitivity Analysis for Nonlinear Programming", Mathematical Programming, Vol. 6, 1974.

2. Sobieski, J., Barthelemy, J.F. and Riley, K.M., "Sensitivity of Optimum Solution to Problem Parameters", AIAA Journal, Vol. 20, Sept. 1982.

3. Schmit, L.A. and Cheng, K.J., "Optimum Design Sensitivity Based on Approximation Concepts and Dual Methods", International Journal for Numerical Methods in Engineering, Vol. 20, 1984.

4. Vanderplaats, G.N. and Yoshida, N., "Efficient Calculation of Optimum Design Sensitivity", AIAA Journal, Vol. 23, No. 11, 1985.

5. Beltracchi, T.J. and Gabriele, G.A., "An Investigation Using an R&P Method to Calculate Parameter Sensitivity Derivatives", Proceedings of NASA Conference, Recent Advances in Multidisciplinary Analysis and Optimization, Hampton, Virginia, Sept. 28-30, 1988.

6. Janin, R., "Sensitivity for Non-Convex Optimization Problems", Convex Analysis and its Applications, Proceedings of a Conference held at Muret-Le-Quaire, March, 1976.

7. Jittorntrum, K., "Solution Point Differentiability Without Strict Complementarity in Nonlinear Programming", Mathematical Programming, Vol. 21, 1989.

Table 1

Right Derivative of the Optimum Values of A_i with Respect to
Allowable Stress in Member 9

Variables	Sensitivities (in^2/psi $\times$ 10^5)			
	Optimal Solution (in^2)	Kuhn Tucker	Quadratic Full	Quadratic Diagonal
A_1	7.936	-0.400	-0.399	-0.405
A_2	0.100	0	0	0
A_3	8.078	0.400	0.399	0.405
A_4	3.929	-0.400	-0.399	-0.405
A_5	0.100	0	0	0
A_6	0.100	0	0	0
A_7	5.758	0.566	0.563	0.573
A_8	5.558	-0.566	-0.564	-0.573
A_9	4.634	-15.932	-15.871	-16.132
A_{10}	0.100	0.570	0.568	0.537

Table 2

Left Derivative of the Optimum Values of A_i with Respect to
Allowable Stress in Member 9

Variables	Sensitivities (in^2/psi $\times$ 10^5)			
	Optimal Solution (in^2)	Kuhn Tucker	Quadratic Full	Quadratic Diagonal
A_1	7.936	0.167	0.149	0.151
A_2	0.100	0	0	0
A_3	8.078	-0.167	-0.175	-0.176
A_4	3.929	0.167	0.172	0.172
A_5	0.100	0	0	0
A_6	0.100	0	0	0
A_7	5.758	-0.235	-0.298	-0.298
A_8	5.558	0.235	0.245	0.247
A_9	4.634	15.656	16.242	16.244
A_{10}	0.100	0	0.064	0.065
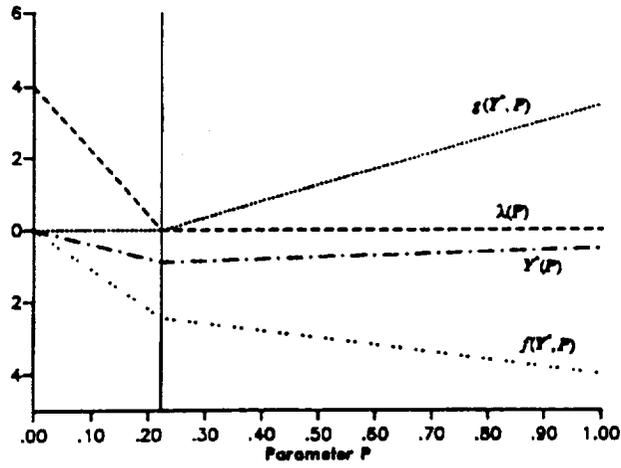
Figure 1. One Design Variable Problem.
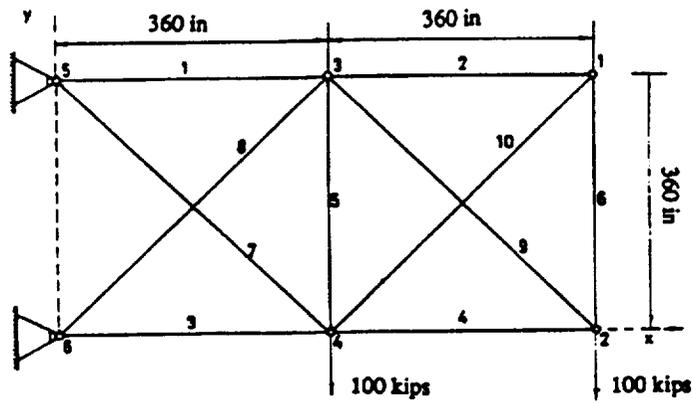
Figure 2. Ten Bar Truss Structure.
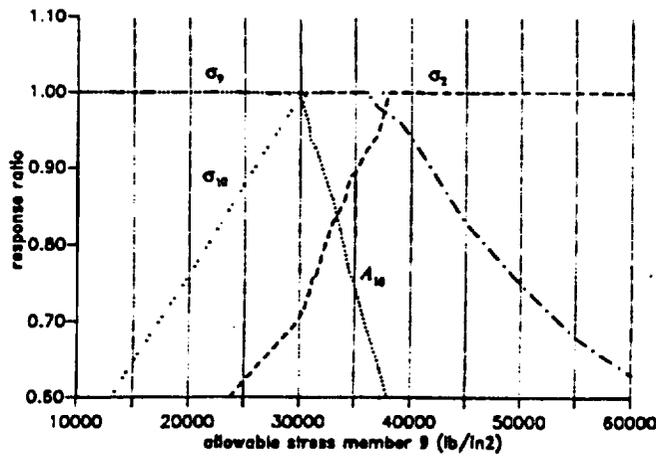
Figure 3. Changes in Active Constraints for Ten Bar Truss Example.

SENSITIVITY ANALYSIS OF THE OPTIMAL SOLUTION OBTAINED FROM THE STRUCTURAL DAMAGE IDENTIFICATION PROCESS

Mo-How H. Shen, Assistant Professor
Department of Aeronautical and Astronautical Engineering
The Ohio State University
Columbus, Ohio 43210-1276

Abstract

An identification procedure proposed by Shen and Taylor [1] to determine the crack characteristics (location and size of the crack) from dynamic measurements is tested. This procedure was based on minimization of either the 'mean-square' measure of difference between measurement data (natural frequencies and mode shapes) and the corresponding predictions obtained from the computational model. The procedure is tested for simulated damage in the form of symmetric cracks in a simply-supported Bernoulli-Euler beam. The sensitivity of the solution of damage identification problems to the values of parameters that characterize damage is discussed. A sensitivity formula is derived.

Introduction

Many optimum or light-weight designed load-carrying structural systems such as tubines, generators, motors, aircraft, and spacecraft are under sever operational conditions. One form of damage that could lead to several failure of the system if undetected is cracking structural member of the system. This motivates the search for new methods of crack monitoring which are not only quicker, and cheaper, but also capable of detecting the integrity of structural members. Most importantly, these crack monitoring schemes could even be performed on line.

An detection procedure was developed by Shen and Taylor [1] to determine the crack characteristics (location, xc, and size, cr, of the crack) from dynamic measurements. The idea of this procedure was related to methods of structural optimization. Specifically, the structural damage is identified in a way to minimize one or another measure of the difference between a set of data (measurements) T_d, and the corresponding values for dynamic response M_d obtained by analysis of a model for the damaged beam. This may be expressed symbolically as the following optimization problem:

$$\min_{xc,cr}\ norm(T_d - M_d). \qquad (1)$$

Naturally, the minimization represented here is constrained by the equations which model the physical system. Moreover, as indicated in the discussion by Shen and Pierre [2-4], one can note that the more modal information used for crack detection, the more accurate and reliable the result that can be achieved. For practical purposes, the objective of Eq. (1) was formulated based on a certain set of specific modes; specifically the first M modes are considered in the inverse procedure.

In this study the corresponding to the mean-square measure of the norm is examined. The identification process is based on minimization of the 'mean-square' measure of difference between measurement data (natural frequencies and mode shapes) and the corresponding predictions obtained from the computational model. The identification procedure is tested for simulated damage in the form of a simply-supported Bernoulli-Euler beam. The uniqueness and reliability of the identification process is confirmed by solving several crack identification examples with specified crack

positions. Without knowing the damaged location, a restricted region in initial data space had been found for which there will be a realistic and convergent solution from the identification process. This region is small, and can be expanded if modal variables are well approximated and initial data corresponding to higher modes of the beam are included in the process. However, for practical reasons, in structural dynamic testing only a small subset of the eigenvalues and eigenvectors can be represented in the measurement data. Futhermore, even if substantially more modal information would be available, the minimization search may be prohibitive for such a large-dimensional feasible domain that would result.

A concept of improving the above identification procedure is also purposed in this study. A sensitivity formula is derived there. Some questions related to the selection of proper modes to be used in the optimization process is also discussed in this study.

A Brief Review of Cracks Identification Procedures [1]

In reference [1], the mean square differences between measured and modeled values of frequency and mode shape are employed as the objective function in the variational formulations for the identification of a cracked beam with one pair of symmetric cracks are presented. In other words, the inverse process seeks to determine the crack parameters, xc and cr, in the mathematical model to minimize the mean square difference between the test data and analytical predictions. In addition, the identification problem was treated as well in the form of a min-max problem in [1]. For simplicity, only the mean-square problem formulation was presented in form consistent with having the beam deflection data stated in discrete form. These problem formulations are

$$\min_{cr,xc} \left(\sum_{\alpha=1}^{M} [(\omega_{t\alpha}^2 - \omega_\alpha^2)^2 + \sum_{m=1}^{T} (w_{t\alpha}(x_{tm}) - w_\alpha(x_{tm}))^2] \right) \tag{2}$$

subject to constraints that define the beam response w_α (ie., the equations for free vibration), and which prescribe appropriate normalization of w_α and test data $w_{t\alpha}$. subject to:

$$\int_0^l \{EIQ(w_\alpha''(x))^2 - \omega_\alpha^2 \rho A \, w_\alpha^2(x)\}dx = 0 \tag{3}$$

$$\sum_{m=2}^{T-1} (w_\alpha(x_{tm})w_\beta(x_{tm}))\Delta x_{tm} - \eta_{\alpha\beta} = 0 \tag{4}$$

$$(cr + \underline{a}\,xc) - R \leq 0 \tag{5}$$

$$\underline{cr} \leq cr \leq \overline{cr} \tag{6}$$

$$\underline{xc} \leq xc \leq \overline{xc} \tag{7}$$

where $\alpha, \beta = 1,...,M$, $\underline{a}$ is a weighting factor on the cr and xc, R represents the upper bound on value $cr + \underline{a}xc$, and $\overline{xc}, \underline{xc}$, and $\overline{cr}, \underline{cr}$ represent the upper and lower bounds of the crack (damage) parameters xc and cr, respectively. Here $cr = \frac{d-h}{d}$ represents crack ratio (a measure of crack depth), and xc identifies crack position (see Fig. 1).

The effect of cracks on the structural properties of the beam is reflected by factor Q in Eq. (3), as described for symmetric surface cracks in Shen and Pierre [2] and in Christides and Barr [5], and for the single surface crack problem in Shen and Pierre [3,4]. In other words, the optimization parameters xc and cr cited in Eq. (2) enter the problem via Q.

66

Numerical Examples

The numerical optimization technique set forth in this study for vibrating cracked beam identification problems is accomplished using the VMCON optimization package program (this implements a sequential quadratic programming method). The damage properties (cr and xc) of the simply supported cracked beams are identified by direct solution of the optimalization problems described in the previous section.

The cracked beam model to which the identification procedure is applied is shown in Fig. 1. It is a simply supported beam of length l equal to 18.11 of it's thickness d, with uniform rectangular cross-section area A, and a pair of symmetric cracks of $cr = 0.5$ located at mid-span ($xc = 0.5$).

Examples with position of the crack (damage) specified

Consider the first example for crack identification, the simply supported cracked beam, for which the crack position xc is known. In other words, only the crack ratio cr is to be identified; therefore, the variables in this problem are $cr, \xi's$, and mode shapes $w_\alpha(x)$ ($\underline{x}_1 = \{cr, \xi_\alpha, w_\alpha(x_{tm})\}$). Furthermore, according to the observations in Shen and Pierre [2,3], the even modes of a simply supported beam are not sensitive to a mid-span crack; therefore, in effect only first and third mode ($\alpha = 1,3$) information is used to represent crack damage.

In Table 1, the top row denotes the assumed crack ratio and corresponding first and third eigenfrequencies. The symbol $*$ denotes the expected optimal solution through the identification process. The first two column entries, ξ_1, ξ_3, indicate the fundamental and the third frequencies corresponding to the initial crack ratio cr which is given in the next column. The last three columns give the final values corresponding to previous entry values. These final values are obtained at the stage where computation is terminated when the further optimal search obtains improvements for criterion F less than the specified tolerance ($10E - 5$ was adopted in the present study). Recall that for an uncracked beam cr is identically zero. Therefore, in this example, it is decided to start with the case of the initial value $cr = 0.0$ and for each case thereafter the cr value is increased by 0.1.

In Table 1, rows 5 to 11 present the results for cases with initial $cr = 0.1$ to 0.8. The corresponding final point values listed in the columns 4-6 show that these cases exhibit, as expected, similar solution characteristics and accuracy. This provides a physical understanding of the geometry of the solution set: for the inverse cracked beam problem with specified crack position, the mean square criterion of Eq. (2) is a convex function and it is bounded by the constraints of Eqs.(3-7). Hence, one may conclude that the convergence of the present optimization problem is obtained independent of the initial data chosen. In other words, as long as the initial data is selected within the problem's feasible domain, an accurate and unique solution through the identification process is expected.

Examples with simultaneous identification of crack position and depth

The second numerical example deals with the crack identification of a simply supported cracked beam with unknown crack ratio and with crack position unknown. In this treatment, the variables in the optimization problem are $cr, xc, \xi's$, and mode shapes $w_\alpha(x)$ ($\underline{x}_1 = \{cr, xc, \xi_\alpha, w_\alpha(x_{tm})\}$).

Table 2 shows that almost all of the cases have unacceptable final estimates of xc and cr. For instance, if the initial position is selected as $xc = 0.4$ and $cr = 0.4$, the values of xc and cr at the final iteration are 0.99789 and 0.36289 which are approximately 98% and 28% different than the given test data. In other words, evidently the configuration with $xc = 0.99789$ and $cr = 0.36289$ is able to provide another minimum value of the criterion (besides the one associated with the expected result). Except for the case with initial $cr = 0.4$ and $xc = 0.48$ which provides less than 1% estimation error. A number of similar examples can be found in reference [1].

Questions arise concerning the conditions under which the identification procedure can pro-

vided a unique solution. As discussed in Shen and Pierre [2-4] and concluded in the studies of Gladwell et. al. [6], if all the mode information is used in the identification procedure, then the system's properties can be identified uniquely. However, for practical reasons, in structural dynamic testing only a small subset of the eigenvalues and eigenvectors can be represented in the measurement data. Futhermore, even if substantially more modal information would be available, the minimization search may be prohibitive for such a large-dimensional feasible domain that would result. These comments are intended to point out certain limitations inherent in the identification procedures.

Sensitivity Analysis of the Optimal Solution from the Damage Idnetification Process

Without using higher modes information, a concept of improving the above identification procedure is presented in this section. The idea is first to obtain the optimal solution of a damage identification process with crack position specified. The actual crack position can then be determined by characterizing the sensitivity of the solution of damage identification process to the value of assumed damage position. In other words, the final solution from damage identification process should be preserved at the new damage position $xc^* = xc + \delta xc$. It is clear that the variables $\underline{x}_1 = \{cr, \xi_\alpha, w_\alpha(x_{tm})\}$ are dependent on parameter xc. From the K-K-T necessary conditions, a set of the equations can be written for new variables x_1^* and new damage position xc^*. In order to achieve an improved solution from damage identification process, δxc can be selected such that the criterion be reduced and constraints be prevented from violations as well. Therefore, to find δxc and x_1^* is equivalent to find a optimal solution of the following problem :

$$\min_{\delta xc} \ (\dot{\Phi}(x_1^*, xc)\frac{db^*}{dx_c} + \Phi'(x_1^*, xc))\delta xc \qquad (8)$$

subject to :

$$[\psi_i + (\dot{\psi}_i\frac{dx_1^*}{dxc} + \psi_i')\delta xc]|_* \leq 0 \qquad (9)$$

$$[\phi_i + (\dot{\phi}_i\frac{dx_1^*}{dxc} + \phi_i')\delta xc]|_* = 0 \qquad (10)$$

where $(\dot{}) \equiv \frac{\partial}{\partial x_1}$, and $(') \equiv \frac{\partial}{\partial xc}$. Functions Φ_i, ψ_i, and ϕ_i are the criterion. inequality constraints, and equality constraints, respectively.

Conclusions

A general method for crack identification of a simple beam with one pair of symmetric cracks is presented. The method may be useful as a component of an on-line nonintrusive damage detection technique for vibrating structures. A variational formulation is expressed as a direct minimization problem statement with a criteria of the mean square difference of natural frequencies and mode shapes between test measurements and corresponding model values. The crack identification problem is reduced to finding the cracked beam's damage parameters that will satisfy appropriate constraints and minimize the mean square difference.

The uniqueness and reliability of the identification process is confirmed by solving several crack identification examples with specified crack positions. Without knowing the damaged location, a restricted region in initial data space has been found for which there will be a realistic and

convergent solution from the identification process. This region is small, and can be improved via the process of sensitivity analysis of the optimal solution from the damage idnetification procedure.

References

[1] M. H. Shen and J. E. Taylor, *J. of Sound and Vibration* submitted for publishion, January 4, 1990. An Identification Problem for Vibrating Cracked Beams.

[2] M. H. Shen and C. Pierre, 1990 *J. of Sound and Vibration* Vol. 138, No. 1, pp. 115-134. Natural Modes of Bernoulli-Euler Beams with Symmetric Cracks.

[3] M. H. Shen and C. Pierre, 1990 *AIAA/ASME/ASCE/AHS 31th Structures, Structural Dynamics and Materials Conference*, Conference Processing, Vol. 4, pp. 2079-2093. Natural Modes of Bernoulli-Euler Beams with a Single-Edge Crack.

[4] M. H. Shen and C. Pierre, *J. of Sound and Vibration* submitted for publishion, May 10, 1990. Free Vibrations of Beams with a Single-Edge Crack.

[5] S. Christides and A. D. S. Barr 1984 *Int. J. Mech. Sci.* 26 No.11/12 639-648. One-Dimensional Theory of Cracked Bernoulli-Euler Beams.

[6] G. M. L. Gladwell, A. H. England, and D. Wang 1987 *J. of Jound and Vibration* 119 81-94. Examples of Reconstruction of an Euler-Bernoulli Beam from Spectral Data.
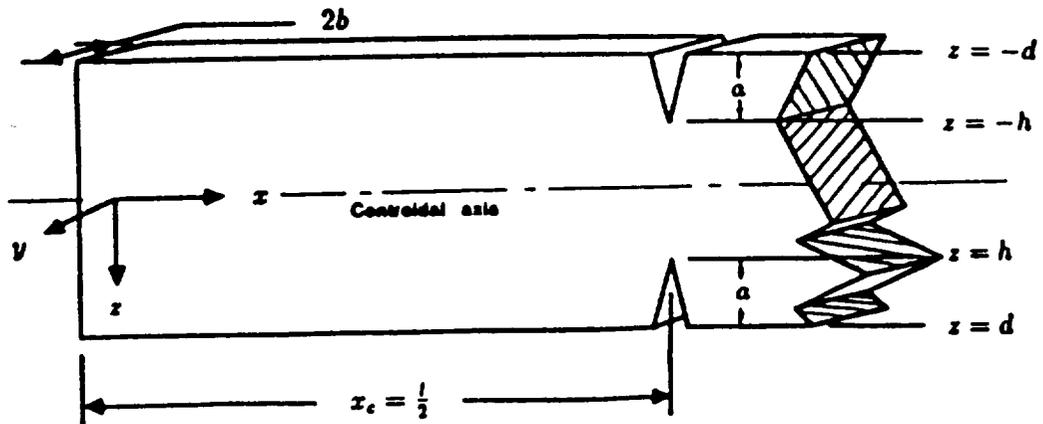
Figure 1. Geometry of a simply supported beam containing a pair of symmetric edge cracks at mid-span. $x_c = \frac{l}{2}$.

Test Data: $\xi_1^*=0.84703$, $\xi_3^*=70.1348$, $cr^*=0.5$					
Initial Data			Final Data		
ξ_1	ξ_3	cr	ξ_1	ξ_3	cr
1.0	81.0	0.0	0.84684	70.1348	0.50033
0.98841	80.0769	0.1	0.84697	70.1346	0.50019
0.97217	78.8135	0.2	0.84704	70.1347	0.49998
0.94815	77.0062	0.3	0.84701	70.1348	0.50007
0.91032	74.3024	0.4	0.84694	70.1347	0.50024
0.73638	63.7848	0.6	0.84705	70.1348	0.49962
0.54574	55.0511	0.7	0.84703	70.1348	0.50034
0.27233	45.9316	0.8	0.84700	70.1347	0.50009

Table 1: Numerical results based on mean square problem statement of Eqs. (2-7) with the crack (damage) specified ($xc = 0.5$).

Test Data: $\xi_1^*=0.84703$, $\xi_3^*=70.1348$, $cr^*=0.5$, $xc^*=0.5$							
Initial Data				Final Data			
ξ_1	ξ_3	cr	xc	ξ_1	ξ_3	cr	xc
0.91806	78.5161	0.4	0.4	0.69639	70.1359	0.99789	0.36289
0.91371	76.6365	0.4	0.43	0.70007	70.1362	0.99440	0.39620
0.91158	75.1335	0.4	0.46	0.84610	70.1347	0.91029	0.53775
0.91056	74.7464	0.4	0.47	0.84711	70.1347	0.67125	0.49033
0.91063	74.5157	0.4	0.48	0.84704	70.1348	0.50554	0.49972
0.73472	63.8062	0.6	0.51	0.84704	70.1348	0.60027	0.50526
0.73711	64.2643	0.6	0.52	0.84704	70.1348	0.60083	0.50531
0.73617	64.7619	0.6	0.53	0.84704	70.1348	0.60141	0.50534
0.73929	65.6727	0.6	0.54	0.84705	70.1348	0.60255	0.49459
0.73909	66.6112	0.6	0.55	0.84702	70.1348	0.99721	0.24709
0.75452	74.0109	0.6	0.6	0.70040	70.1363	0.99079	0.59307

Table 2: Numerical results based on mean square problem statement of Eqs. (2-7). The position of the damage xc is a variable.

C-2.

SIMPLIFIED DETECTION AND CORRECTION OF CRITICAL DATA FOR ILL-CONDITIONED SYSTEMS

By Irving U. Ojalvo

Bullard Professor of Mechanical Engineering
University of Bridgeport, Bridgeport, Connecticut 06601

ABSTRACT

Ill-conditioned systems arising in analysis and optimization can display a high sensitivity to numerical precision for changes and errors in data input. Such data may be in the form of system parameter input or desired system response. The ill-conditioning we refer to generally arises from the lack of sufficient independent data to define a complex system or the weak sensitivity of response to source input parameters. In this paper we shall show how small errors in data and assumed fixed and known parameters can lead to highly erroneous results in ill-conditioned linear algebraic equations. A simplified detection and correction of critical input data arising in the coefficient matrix and desired response (i.e., right hand side) is proposed herein.

INTRODUCTION

Ill-conditioning is characterized by high sensitivity of solutions to small changes in system parameters and data inconsistencies. In this paper we shall trace response data and input parameter errors, and show how they can create significant solution contamination for ill-conditioned systems.

We will then show how a simple detection of these inconsistencies is possible from an examination of the zero and near-zero eigenvalues and eigenvectors of the system. This will be followed by a correction procedure which can be used to identify the sensitive data and to correct it as well as indicate how to correct the system response or design parameters to achieve reasonable solutions.

TECHNICAL APPROACH

A technical statement of the problem is as follows: Determine the parameters, $\{r\}$, which will deliver a desired performance or response, $\{Y\}$, starting with a design based on initial parameters, $\{r_0\}$, which deliver a non-optimum response, $\{Y_0\}$.

Assuming the elements of $\{r\}$ are reasonably close to the desired design, a Taylor series expansion yields

$$\{Y\} = \{Y_0\} + [S]\{r - r_0\} + \{R\} \tag{1}$$

where $\{R\}$ are the remainder terms in the approximation and $[S]$ is the response sensitivity, given by

$$[S] = \left[\frac{\partial\{Y\}}{\partial\{r\}} \right]_{r = r_0} \tag{2}$$

The usual linearization procedure employed for such problems involves performing a least error-squared minimization of the remainder terms which results in the symmetric formulation

$$[A]\{x\} = \{b\} \tag{3}$$

where

$$[A] = [S^T S]$$

$$\{b\} = [S]^T \{Y - Y_0\} \tag{4}$$

72

$$\{ x \} = \{ r - r_o \}$$

If there is insufficient independent performance data $\{ Y - Y_o \}$ to uniquely determine the system parameters, $\{ r \}$, $[A]$ will be ill-conditioned and a solution of Eqn. (3) for the system parameters $\{ r \}$ will be difficult to achieve.

One procedure for solving Eqn. (3) if $[A]$ is ill-conditioned (i.e., $[A]$ has zero or near-zero eigenvalues) is to use the Singular Value Decomposition (SVD) method.[1] This requires that the analyst obtain all the eigenvalues and eigenvectors of $[A]$, which can be computationally expensive if the size of $[A]$ is large (e.g., 300 by 300) even when the rank deficiency of $[A]$ is small.

A more efficient procedure is to use Epsilon-Decompositions[2] (E-D) which only requires computing the zero, near-zero and lowest non-small eigenvalues. This is reasonable since it is only the zero and near-zero eigenvalues which make the system ill-conditioned. In this paper we will show how to detect and eliminate critical data errors and parameter contamination errors.

CHARACTERIZATION OF ERRORS

Let us replace Eqn (3) by its approximate version, i.e.,

$$A' \ x' = b' \tag{5}$$

where we have dropped the brackets surrounding the matrices and vectors

$$A' \approx A$$
$$b' \approx b$$

but $x' \neq x$ since A is ill-conditioned.

The approximations in A' are caused by system parameter and sensitivity errors, whereas the approximations in b' are due to response errors.

Let ϕ_i be the normalized eigenvectors of A and λ_i be its corresponding eigenvalues. Thus,

$$A \Phi = \Phi \Lambda$$

$$\Phi^{-1} = \Phi^T \tag{6}$$

73

where
$$\Phi = [\phi^1, \phi^2, \dots \phi^n]$$

and

$$\Lambda = \begin{bmatrix} & & \\ & \lambda_i & \\ & & \end{bmatrix}$$

The approximation to the eigenvalue matrix, Λ is Λ'
where

$$\Lambda' = \Lambda + P$$

i.e., $\lambda_i = \lambda'_i + p_i$

We shall assume that

$$\Phi' \approx \Phi$$

Therefore,
$$\Phi^T [A'] \Phi \approx \Lambda' \tag{7}$$

Letting
$$x' \approx \Phi c'$$
$$b' \approx \Phi \beta' \tag{8}$$

and substituting Eqn (5) and (6) into Eqn (4) yields

$$(\lambda_i + p_i) c_i' = \beta_i' \tag{9}$$

where we have made use of Eqn (5) and (6).

Subtracting $\lambda_i c_i = \beta_i$ from Eqn (9) and rearranging terms yields

$$c_i' - c_i = \left[\frac{\beta_i' - \beta_i}{\lambda_i} \right] - \left[\frac{p_i c_i'}{\lambda_i} \right] \tag{10}$$

EFFECTS OF ILL-CONDITIONING

If the matrix A' is ill-conditioned then there will be one or more very small eigenvalues, λ'_i. Therefore, errors in b', as reflected by $\beta_i' - \beta_i$, can have a significant influence on errors in x, as reflected by $c_i' - c_i$. This can be easily seen from Eqn. (10).

Similarly, errors in the sensitivities and input parameters, as characterized by the p_i, can also influence errors in x (or $c_i' - c_i$).

SIMPLIFIED CORRECTION OF DATA SENSITIVITY

Since the large error source in x, for small errors in A and b, are caused by the very small eigenvalues, λ'_i, it is proposed that the ϕ_j associated with these λ_j be computed and their contributions to A' and b' be removed as follows:

$$b' - b \approx \Sigma \alpha_j \phi_j \qquad (11)$$
where

$$\alpha_j = \phi_j^T b'$$
and

$$A' - A \approx \Sigma \lambda_j' [\phi_j \phi_j^T] \qquad (12)$$

Thus, the A' parameter matrices and b' response vectors are changed only as they affect the ill-conditioned nature of the problem. Once these critical changes are proposed, the analyst must decide if these potentially sensitive changes should be made consistent with the potential error sources in the data obtained.

One may also wish to trace back the proposed change in A' and b' to $[S']$ and $\{Y' - Y'_o\}$ using Eqn. (4).

CONCLUSION

A simplified detection and correction of critical data sensitivity for ill-conditioned systems has been proposed. It requires computation of the zero and near-zero eigenvalues, an their corresponding eigenvectors, for the system at hand. Only a theoretical derivation has been presented. Numerical examples will be treated in future investigations.

REFERENCES

1. Maia, N.M.M., "An Introduction to the Singular Value Decomposition Technique (SVD)," 7th IMAC, 1989, Las Vegas, NV, Proceedings Vol. 1 (pp. 335-339).

2. Ojalvo, I.U. "Improved Solutions for System Identification Equations by Epsilon Decomposition," 31st SDM Conference, April 1990, Long Beach, CA (Paper No. 90-1148).

A Comparison of Optimization-Based Approaches for

Solving the Aerodynamic Design Problem

Paul D. Frank
Gregory R. Shubin

Boeing Computer Services
P.O. Box 24346, MS 7L-21
Seattle, WA 98124-0346
June, 1990

Abstract

In this paper we compare three optimization-based methods for solving aerodynamic design problems. We use the Euler equations for one-dimensional duct flow as a model problem, and compare the three methods for efficiency, robustness, and implementation difficulty. The smoothness of the design problem with respect to different shock-capturing finite difference schemes, and in the presence of grid refinement, is investigated.

1. Introduction

Most of the effort in devising schemes for solving computational aerodynamics problems has focused on the forward, or *analysis* problem: given the shape of the airfoil (or aircraft), what will be the flow of air over it? Of more direct use in designing an aircraft is the solution of the more difficult inverse, or *design* problem: given the flow, what shape will produce it? Recently, due to improved methods for solving the analysis problem, and also due to increases in available computational power, there has been renewed interest in attacking the design problem.

Many different approaches to solving the design problem have been developed; these are nicely summarized in [1]. For our purposes, these approaches can be separated into two fundamental classes. In the first class, one attempts to solve the inverse problem by (essentially) manipulating the equations governing the geometry and the flow so that the geometry can be solved for, once the flow is specified. In the second class, a method for solving the forward problem is used iteratively, employing an optimization strategy to vary the airfoil shape in some systematic way until (close to) the desired flow is obtained. The second class of methods, while generally much more computationally intensive than the first, offers more promise for handling difficult geometries and complex flow phenomena, and takes advantage of existing methods for solving the associated analysis problems.

The objective of this paper is to compare several optimization-based approaches for solving the design problem. To do so, we introduce a very simple model problem. The analysis problem for this model is well-known and has been widely used for testing numerical methods for flows with shocks; it is the problem of determining the steady, one-dimensional flow of an inviscid fluid in a duct with a specified spatially-variable cross-sectional area. The design problem for the model is to determine the duct shape from the flow solution. Except for one-dimensionality, the flow phenomena exhibited by solutions of the model are quite similar to those in two-dimensional inviscid flow over an airfoil; this point is illustrated in Figure 1. Thus, we may hope to gain some insight into the nature of the airfoil design problem by studying the vastly simpler duct flow model.

In Section 2 below we present the model analysis and design problems. In Section 3 we present three distinct optimization methods for solving the design problem, and discuss the relationships between them. In Section 4 we display some computational results using the three methods, and discuss the tradeoffs between them. In Section 5 we present our conclusions.

2. Model Problem

2.1 Continuous Analysis Problem

The steady flow of an inviscid fluid in a duct of variable cross-sectional area $A(\xi)$ is governed by the Euler equations

$$\mathcal{F}_\xi + \mathcal{G} = 0, \qquad 0 \le \xi \le 1, \tag{1}$$

where

$$\mathcal{F} = \begin{pmatrix} \rho u A \\ (\rho u^2 + p)A \\ (\rho E + p)u A \end{pmatrix}, \qquad \mathcal{G} = \begin{pmatrix} 0 \\ -pA_\xi \\ 0 \end{pmatrix},$$

ξ is distance along the duct, ρ is density, u is velocity, $E = e + u^2/2$ where e is specific internal energy, and p is pressure. Here, the subscript ξ means differentiation with respect to ξ, and it is assumed that $A(\xi)$ is a given, differentiable function. The pressure p is given by the equation of state for a perfect gas, $p = (\gamma - 1)\rho e$, where $\gamma > 1$ is the gas constant. (For air, $\gamma = 1.4$.) We assume supersonic inflow at $\xi = 0$ and subsonic outflow at $\xi = 1$. Under these circumstances, it is proper to specify three boundary conditions at $\xi = 0$ and one boundary condition at $\xi = 1$ [2].

In [3] we show how, under these conditions, (1) can be reduced to a single ordinary differential equation in u. This equation is

$$f_\xi + g = 0 \tag{2}$$

where

$$f(u) \equiv u + \bar{H}/u, \qquad g(u,\xi) \equiv \frac{A_\xi}{A}(\bar{\gamma} u - \bar{H}/u),$$

and $\bar{\gamma} = (\gamma - 1)/(\gamma + 1)$ and $\bar{H} = 2H\bar{\gamma}$ are given constants. Here, $H = \gamma e + u^2/2$ is the total enthalpy, which is evaluated at the inflow boundary. Equation (2) is fully equivalent to (1); no approximations have been made in the derivation.

Now we pose our analysis problem, specified so that the solution contains a single shock at ξ_s, is supersonic for $0 < \xi < \xi_s$, and subsonic for $\xi_s < \xi < 1$.

Analysis Problem
   Given:

$$A(\xi), \quad A_\xi > 0 \tag{3a}$$

   Find:

$$u(\xi) \quad \text{satisfying} \quad \begin{cases} f_\xi + g = 0, & \text{away from the shock;,} \\[2mm] u_L \cdot u_R = \bar{H} \text{ and } u_L > u_\ast > u_R, & \text{at the shock;} \\[2mm] u(\xi = 0) = u_{in} > u_\ast \\[2mm] u(\xi = 1) = u_{out} < u_\ast \\[2mm] \text{and other technical conditions} \end{cases} \tag{3b}$$

Here, $u_\ast$ is the sonic velocity $\sqrt{\bar{H}}$, the conditions at the shock are the Rankine-Hugoniot jump relation and the entropy condition, and the specified boundary values are the inflow and outflow conditions. The technical conditions amount to certain relationships between u_{in} and u_{out} that must hold in order for a solution to exist; see [3].

78

2.2 Discrete Analysis Problem

We introduce three discretization methods for (2) to be used in solving (3) ; these methods differ in their degrees of continuity, which has an effect on the results obtained with the design optimization methods presented later.

Let the ξ -coordinate be discretized by a uniform, cell-centered grid with centers at $\xi_j = (j - 1/2)h$, $h = 1/J$, where J is the number of unknown grid values. Let u_j represent a piecewise constant approximation to u on each grid cell. Then, a conservative difference scheme for (2) is given by

$$w_j \equiv \frac{f_{j+1/2} - f_{j-1/2}}{h} + g_j = 0. \tag{4}$$

Here the source term $g_j = g(u_j, (A_\xi/A)_j)$ and we assume that the duct shape $A(\xi)$ is given by a piecewise cubic spline described in the B-spline basis [4] with coefficients b_m for $m = 1, 2, \ldots, M$ and that $A(0)$ and $A(1)$ are fixed. $(A_\xi/A)_j$ is obtained by evaluating the spline and its derivative at ξ_j. The boundary conditions on u are $u_0 = u(\xi = 0)$ and $u_{J+1} = u(\xi = 1)$.

It remains to prescribe the fluxes $f_{j+1/2}$ as functions of u_j and u_{j+1}. Three such prescriptions are f^G, f^{EO}, and f^{AV}, corresponding to the Godunov, Engquist-Osher, and Artificial Viscosity methods for numerically approximating hyperbolic conservation laws; see [3]. The Godunov flux f^G corresponds roughly to the first order upwind scheme frequently used in computational aerodynamics, and is a C^0 function of its arguments. The Engquist-Osher flux f^{EO} is a slight perturbation of f^G that makes it C^1. The artificial viscosity flux f^{AV} is entirely different, and is C^∞. The abilities of these schemes for sharply representing computed shock waves vary somewhat inversely to the degree of continuity, with the Godunov scheme having about one grid cell interior to a shock, the Engquist-Osher scheme two cells, and the Artificial Viscosity scheme many cells. Because continuity is an issue later, we will refer to these schemes as the C^0, C^1, and C^∞ difference schemes, respectively.

Once the discretization has been made, we are faced with solving a system of nonlinear algebraic equations. The system is

Discrete Analysis Problem
Given: b_m, $m = 1, \ldots, M$ (spline coefficients describing $A(\xi)$)
Find: u_j satisfying

$$W(u) = 0. \tag{5}$$

Here W is the vector of discretized equations (4) for $j = 1, 2, \ldots, J$ and the boundary conditions on u. We will refer to the method for solving the analysis problem as the *analysis code*. The actual method employed in the analysis code may be Newton's method (or a variant), some other iterative method (e.g., multigrid), or a time-marching scheme that approximates a time-dependent differential equation.

2.3 Continuous Design Problem

We next turn our attention to posing the design (or inverse) problem: given the flow solution $u(\xi)$, what is the duct geometry $A(\xi)$? In other words, we want to find that duct geometry $A(\xi)$ such that the solution of (3) is some specified function $\hat{u}(\xi)$. Some of the obvious ways of stating the design problem are mathematically improperly-posed; see [3]. Given these difficulties, we are led naturally to seek a least squares approximate solution; that is, we want to solve

Design Problem
Given: $\hat{u}(\xi)$
Find: $A(\xi)$, $A_\xi > 0$ such that $u(\xi)$ satisfies (3b) and $\|u(\xi) - \hat{u}(\xi)\|_2$ is minimized.

We note that this particular objective function puts a large premium on getting the shock located correctly, and that precise location of shocks may not be as important in practical design problems for airfoils or aircraft.

2.4 Discrete Design Problem

We assume that a desired (or goal) velocity distribution $\hat{u}_j$ is given for each computational cell in the analysis problem. Then we have

Discrete Design Problem
  Given: $\hat{u}_j, \quad j = 1, \ldots, J$
  Find: $b_m, \quad m = 1, 2, \ldots, M$ (spline coefficients describing $A(\xi)$) such that (5)
    is satisfied and $\frac{1}{2} \sum_{j=1}^{J} (u_j - \hat{u}_j)^2$ is minimized.

Later we will consider three variations on this problem that amount to leaving $A(\xi)$ unconstrained, requiring $A_\xi > 0$, and requiring $A_{\xi\xi}$ have the "correct" sign. The latter two translate into simple linear constraints on the B-spline coefficients b_m.

3. Approaches to formulating design problems using optimization

Most of the recent literature on the aerodynamic design problem features specific optimization approaches or specific design problems. In this section we present a general view of the problem of optimal design. In particular, we consider three different methods for formulating the design problem as an optimization problem. Concepts for these approaches are illustrated by their application to the duct design problem discussed in Section 2. Except for these illustrations, this section is independent of the previous material.

3.1 The Black-Box Method

The *black-box* method is the most direct approach to optimal design. In the black-box method the analysis code is repeatedly invoked as the design variables are altered by the optimization code. Since the analysis code is independent of the optimization code, it may be treated as a black-box.

If the design is characterized by a vector x_D of n_D *design variables* then the optimal design problem is given by

$$\text{minimize } \mathfrak{f}(x_D) ,$$
$$x_D \in \mathrm{R}^{n_D}$$
$$\text{subject to } C(x_D) \geq 0 , \tag{6}$$

where $\mathfrak{f}(x_D)$ is the objective function* and $C(x_D)$ is a vector of m_D constraint functions. In the black-box method, each evaluation of $\mathfrak{f}(x_D)$ requires a solution by the analysis code.

Often, the function $\mathfrak{f}$ will be formulated in terms of *flow variables* x_F. The flow variables are the physical variables on the discretization grid, such as velocities or pressures. For example, the objective for the duct design problem is a function of velocities on the grid cells. In this situation, $\mathfrak{f}$ is dependent on the design variables x_D in an indirect manner. That is, the variables x_D are linked to the flow variables x_F via the differential equations or the discretization of these equations, since the flow variables will change when (for example) the geometry is altered. In general, $\mathfrak{f}$ will have both a direct dependence on x_D and an indirect dependence on x_D, due to the dependence of x_F on x_D. Thus, one could consider the objective function to be $\mathfrak{f}(x_F(x_D), x_D)$. The term $x_F(x_D)$ indicates that, given x_D, the value of x_F is obtained by solving an analysis problem.

The constraints C may also have an indirect dependence on the design variables. However, they will often be shape constraints formulated directly in terms of x_D. For example, one version of the duct design problem requires that the duct area increase monotonically. This constraint can be formulated in terms of the coefficients of the piecewise polynomials that define the area function $A(\xi)$.

One of the drawbacks of the black-box approach is high computational cost. Typically, efficient optimization codes (see e.g., [5]) for solving (6) require computation of $\nabla_D \mathfrak{f}$ and $\nabla_D C$, the gradients of the objective function and constraints with respect to the design variables. Computing these derivatives

---

 * Note that the typewriter font $\mathfrak{f}$ is used for the objective function to distinguish it from the flux function f introduced in Section 2.

by one-sided finite differences requires solving n_D analysis problems, where each problem corresponds to a perturbation of a different component of x_D. One mitigating factor is that solving these perturbed analysis problems should be considerably cheaper than solving arbitrary problems, at least when the analysis code employs an iterative solver. This is due to the availability of the solution of the 'nearby' problem at the nominal value of x_D as a starting guess for the iteration at the perturbed value of x_D. In the next section we show that the first derivatives for the design problem can often be computed by another method for considerably less cost than solving n_D analysis problems.

An advantage of the black-box approach is that the analysis code can be used essentially without modification. Thus, there is no need to tamper with complicated discretization schemes such as those used in most advanced computational aerodynamics codes.

3.2 A Black-Box Scheme Using an Implicitly-Derived Gradient

In this section we describe a method, based on the implicit function theorem, for "cheaply" computing derivatives required in the optimization. Similar methods are mentioned in reference [6] and the citations therein. For simplicity, the unconstrained version of (6) is considered. However, the results apply to the constrained problem as well.

Assume that the analysis problem has been discretized so that an analysis consists of solving a system of nonlinear equations. In this case function evaluations for the black-box method are computed as follows. Given a design specified by x_D, the analysis code solves $W(x_F) = 0$, where x_F is the vector of n_F flow variables and W is a vector of n_F nonlinear equations. Since the analysis problem is an implicit function of x_D it can be viewed as solving

$$W(x_F, x_D) = 0 \tag{7}$$

for x_F, given a design specified by x_D.

Suppose that x_F and x_D are considered as subsets of the $n_F + n_D$ vector x given by

$$x \equiv (\quad x_F \quad | \; x_D); \tag{8}$$

the Jacobian (first-derivative) matrix of (7) is then

$$J = \left[\quad J_F \quad \middle| \; J_D \right] , \tag{9}$$

where J is $n_F \times (n_F + n_D)$, J_F is the $n_F \times n_F$ Jacobian with respect to the flow variables and J_D is the $n_F \times n_D$ Jacobian with respect to the design variables. (The partitioned view of the Jacobian implies $n_F \gg n_D$; this will usually be the case.) Note that J_F is often available in analysis codes, especially those based on Newton's method and variants.

Consider the function $\tilde{\imath}(x_F, x_D)$, where $\tilde{\imath}$ is the same as the black-box method objective function $\imath$, except that x_F and x_D are considered to be independent of each other. The function $\tilde{\imath}(x_F, x_D)$ is then equivalent to the black-box method objective function $\imath(x_F(x_D), x_D)$ only when (7) is satisfied. Computing gradients of $\tilde{\imath}$ is considerably simpler than computing gradients of $\imath$. This is due to the fact that the partial derivatives of $\tilde{\imath}$ with respect to variables in x_D can be computed with the assumption that x_F is fixed. In contrast, the partial derivatives of $\imath$ with respect to variables in x_D must account for the fact that x_F is a function of x_D.

Usually $\nabla_F \tilde{\imath}$ and $\nabla_D \tilde{\imath}$, the gradients of $\tilde{\imath}$ with respect to the flow variables and the design variables, respectively, are available as an analytic expressions or can easily be computed by finite differences. For example, the discrete design problem for duct flow has $(\nabla_F \tilde{\imath}(x))_j = u_j - \hat{u}_j$ and $\nabla_D \tilde{\imath}(x) = 0$. However, the black-box method requires $\nabla_D \imath$, the gradient of $\imath$ with respect to the design variables x_D. The theorem below provides an efficient way to compute $\nabla_D \imath$, given $\nabla_F \tilde{\imath}$ and $\nabla_D \tilde{\imath}$; the proof may be found in [3].

Theorem *If $W(\bar{x}_F, \bar{x}_D) = 0$ and $W(x_F, x_D)$ is C^1 in a neighborhood of $\bar{x} = (\bar{x}_F, \bar{x}_D)$, with J_F nonsingular at $\bar{x}$ then*

$$\nabla_D \imath(\bar{x}_D) = \nabla_D \tilde{\imath}(\bar{x}) - J_D^T J_F^{-T} \nabla_F \tilde{\imath}(\bar{x}) . \tag{10}$$

(Here, superscript T indicates transpose.)

The following algorithm could be used for computing $\nabla_D \tilde{f}$ using equation (10). First compute $\nabla_F \tilde{f}$ and $\nabla_D \tilde{f}$, solve $J_F^T y = \nabla_F \tilde{f}$ for y and then compute $\nabla_D \tilde{f} = \nabla_D \tilde{f} - J_D^T y$. Thus, computation of $\nabla_D \tilde{f}$ by the implicit method requires computing J_D and solving the linear system $J_F^T y = \nabla_F \tilde{f}$. Computation of J_D by forward finite differences requires n_D evaluations of $W(x_F, x_D)$. Note that *evaluation* of $W(x_F, x_D)$ (sometimes referred to as "computing the residuals") is usually significantly cheaper than *solving* $W(x_F, x_D) = 0$, i.e. solving the analysis problem. Solving $J_F^T y = \nabla_F \tilde{f}$ is trivial if the analysis code computes a factorization of J_F. However, if an iterative method such as pre-conditioned conjugate gradient is used in the analysis code, then the iterative solver must be adapted to solve the transposed system.

Some analysis codes do not provide J_F or an iterative solver for systems involving J_F; an example is a time-dependent code where the steady state solution is found by stepping through time. The implicit gradient scheme can still be used in this case, provided that J_F can be computed efficiently using sparse finite differences (see e.g., [7]). The sparse difference approach only requires that the analysis code provide the values of $W(x_F, x_D)$ when values of x_F and x_D are input; most codes, if not already in this form, can be easily modified to produce W.

In general, computing implicit gradients is much cheaper than computing gradients by finite differences. This is because the finite difference gradient computation requires the solution of n_D analysis problems. In contrast, computing the gradient implicitly requires n_D evaluations of the flow equations $W(x_F, x_D)$ and one solve of a linear system with the matrix J_F^T. A disadvantage of the implicit scheme is that some (perhaps substantial) modification of the analysis code is required.

3.3 The All-at-once Method

In deriving the implicit gradient method the objective function $\tilde{f}$ and the discretized differential equations W were considered to be functions of the independent sets of variables x_D and x_F. Thus, one could consider a design method where both x_D and x_F are treated as optimization variables and the flow equations $W(x_F, x_D) = 0$ are treated as equality constraints. This *all-at-once* method can be described formally as

$$
\begin{aligned}
&\text{minimize } \tilde{f}(x_F, x_D) , \\
&x \in \mathrm{R}^{(n_F + n_D)} \\
&\text{subject to } C(x_F, x_D) \geq 0 , \\
&\qquad\qquad W(x_F, x_D) = 0 ,
\end{aligned}
\tag{11}
$$

where $x = (x_F, x_D)$ and the vector C are the design constraints as in (6) . An iteration of the optimization now involves simultaneous modification of both x_F and x_D. A similar approach to the design problem is described in [8].

An advantage of the all-at-once method over the black-box method is the probability of requiring considerably fewer equivalent solutions of the large discretized system $W(x) = 0$. This is because the black-box method requires the solution of $W(x_F) = 0$ for each change in x_D. However, in the all-at-once method, each change in x_D requires the computational equivalent of only one step of a Newton solver for $W(x_F) = 0$.

Another advantage of the all-at-once approach is that it does not require the existence of solutions to the analysis problem for all values of the design variables generated in the course of the optimization. All that is required is that the residual of the system $W(x_F, x_D)$ be computable for the values of x_F and x_D generated by the optimizer. However, by definition, the analysis problem must be analyzable at the optimal value for x_D.

A big disadvantage of the all-at-once method is that the optimization code is not isolated from the analysis code. That is, since the optimization code must simultaneously change the analysis and design variables, it must contain all the specialized software required for an analysis. In particular, even if the number of design variables is small, the optimizer must include code for handling large analysis problems; for example, sparse matrix factorization codes or codes that compute preconditioners and conjugate gradient iterations. Consequently, the all-at-once optimization code may have to be modified significantly for application to each new analysis problem.

Another disadvantage of the all-at-once method compared to the black-box method was discovered in tests on the duct design problem: the all-at-once method is much more susceptible to derivative discontinuities arising from finite difference schemes designed to sharply approximate shocks. This is because the optimization in the black-box method sees potential discontinuities only if shocks move to different grid cells from one converged analysis to the next. In contrast, the all-at-once method has potential derivative discontinuities if shocks move to different grid cells for consecutive values of x_F in the optimization iteration. Since shock locations can move significantly when x_F is far from an analysis solution, this type of discontinuity is much more pervasive with the all-at-once approach.

4. Numerical Results

In this section we present numerical results obtained by applying the design methods discussed in Section 3 to the discrete design problem for duct flow described in Section 2. The testing was done on a Sun SPARC workstation. The optimization code used was NPSOL, a product of the Systems Optimization Laboratory, Stanford University. NPSOL is an implementation of a sequential quadratic programming method.

The design variables (called x_D in Section 3) were the B-spline coefficients describing the duct geometry $A(\xi)$. The two end values of A were fixed at $A(0) = 1.05$ and $A(1) = 1.745$. The tests were run for the case of two design variables ($n_D = 2$) and ten design variables ($n_D = 10$). The results for $n_D = 2$ are found in [3], and those for $n_D = 10$ presented here in some detail. The linear duct shown in Figure 2a was the initial design for each run.

Velocities along the duct were the flow variables (called x_F in Section 3) for the duct design problem. We took $J = 40$ grid cells, so there were $n_F = 40$ flow variables. The boundary conditions were $u_0 = 1.299$ and $u_{41} = 0.506$. The flow variables resulting from an analysis of the linear duct, using the C^0 difference scheme, appear as crosses in Figure 2b.

The analyses in all the black-box method optimization runs were "warm started." That is, the initial values for the flow velocities were taken from the preceding analysis. The initial velocity profile for the first analysis in an optimization run was a linear profile connecting the boundary conditions.

The velocities $\hat{u}_j$ used as the design goal were the evaluations on the computational grid of the analytic solution for a duct with a cross-sectional area given by a sinusoidal perturbation of the linear duct. This velocity profile is the continuous curve in Figure 2b.

Figures 3-5 show the optimal solutions for the $n_D = 10$ duct design problem using the C^0, C^1 and C^∞ difference schemes, respectively. The $n_D = 10$ case allows enough degrees of freedom for "wavy" ducts to be generated in the optimization process. It is clear from Figures 4 and 5 that strangely shaped optimal ducts result from the higher continuity difference schemes that allow a "smearing" of the shock. This is particularly true for the C^∞ scheme.

Table 1 gives the numerical results for the $n_D = 10$ design problem with no constraints. "Bbox (fd grad)" and "Bbox (impl grad)," respectively, denote the black-box scheme with finite difference gradients and gradients computed using the implicit method. "Opt. Found? – Yes" indicates that the optimization code converged to the optimal solution (all optimization methods converged to the same solution for a given difference scheme). The number of optimization iterations, number of function evaluations and CPU time are indicated in the "No. Itrns," "No. Fevals," and "Time" columns, respectively. The number of gradient evaluations is approximately the same as the number of function evaluations. The "No. Equiv. Newton Steps" column indicates the number of times the optimization method requires a computation that is is equivalent to the work of a Newton step on the discretized analysis problem, solve $W(x_F) = 0$. This measure of computation cost is used because this cost will dominate for large problems. Inclusion of equivalent Newton step results is intended to provide a more meaningful basis for performance evaluation than would be obtained by solely considering CPU times on a small problem.

The black-box scheme using implicit gradients is always more efficient than the black-box scheme with finite-difference gradients. The advantage of using the implicit gradient scheme increases as the number of design variables increases. This is to be expected since n_D is the number of analyses required to compute the gradient by one-sided finite differences, whereas n_D is only a secondary factor in the computation cost for the implicit gradient method.

In results not shown here, we found the all-at-once method to be susceptible to difficulties due to low-continuity finite difference schemes; sometimes, it would not converge, or would converge to an undesirable

Table 1. Test Results for $n_D = 10$, no constraints.

Problem Formulation	Difference Scheme	Opt. Found?	No. Itrns	No. Fevals	No. Equiv. Newton steps	Time (secs)
Bbox (fd grad)	C^0	yes	31	48	1216	105.4
Bbox (impl grad)	C^0	yes	31	48	317	35.9
all-at-once	C^0	yes	19	30	19	32.3
Bbox (fd grad)	C^1	yes	30	46	1143	97.1
Bbox (impl grad)	C^1	yes	28	45	315	27.2
all-at-once	C^1	yes	19	31	19	33.8
Bbox (fd grad)	C^∞	yes	28	40	998	68.9
Bbox (impl grad)	C^∞	yes	29	42	257	23.3
all-at-once	C^∞	yes	11	12	11	18.2

local minimum. However, when the all-at-once method does work, as in the C^∞ case, it is much more efficient than the black-box schemes. This is particularly true in terms of equivalent Newton steps.

To make the optimal duct design more physically reasonable, design constraints were imposed. Initially, first-derivative positivity (monotonicity) constraints were imposed. The optimal duct for the C^0 scheme is unaffected by the monotonicity constraint. The monotonicity constraint yields an acceptable optimal duct for the C^1 scheme. The optimal duct for the C^∞ scheme with monotonicity constraints is considerably smoother than the optimal unconstrained duct, but it is still somewhat ugly.

In addition to the monotonicity constraints, second-derivative constraints were imposed which required the duct curvature to have the "correct" sign. The second-derivative constraint has little effect on the optimal C^0 and C^1 scheme ducts. However, the new constraint results in an acceptable optimal design for the C^∞ scheme.

Table 2 compares the number of equivalent Newton steps required for the all-at-once method with those required for the most efficient black-box method, on problems where they both computed the optimal design. The all-at-once method has a significant advantage in the unconstrained case. However, adding design constraints reduces this advantage. A partial explanation of this trend is that the optimization method solves a quadratic programming problem (QP) at every iteration. Since the all-at-once method includes the flow variables in the optimization problem, it works with a much larger QP than the black-box methods. Thus, when constraint inequalities enter and leave the active set, the all-at-once method must perform linear algebra computations on much larger problems than the black-box methods. Despite this disadvantage, the all-at-once method always required many fewer equivalent Newton steps on C^∞ scheme problems than the black-box methods.

Based on the duct design tests, several summary statements can be made. A general trend is that increasing the continuity of the difference scheme reduces the difficulty in the optimization runs, but increases the degree to which the design must be constrained. Summary observations comparing the three problem formulations are given in Table 3. They are compared based on robustness, computational cost and the extent to which they allow independence of the optimization and analysis codes. (The two black-box methods tied for first in the robustness category.)

The test results indicate the desirability of improving the robustness of the all-at-once method so that its efficiency advantage can be exploited. One way to do this is to give the all-at-once method a very good initial estimate of the solution for both the flow and design variables. This idea was tested on the $n_D = 10$ design problem, using the C^0 difference scheme, with both monotonicity and curvature design constraints. The initial flow and design variables were taken from the optimal solution computed by the all-at-once method on the C^∞ version of this problem. The all-at-once method then converged to the optimal solution of the C^0 problem at the cost of 143 equivalent Newton steps. The total cost for both the C^∞ initial solution and the final run on the C^0 problem was 198 equivalent Newton steps. This is an improvement over the 315 equivalent Newton steps required by the best black-box method.

The final test results presented here relate to the smoothness of the optimal design problem. For this

Table 2. Equivalent Newton Steps for Bbox (impl grad) *vs* All-at-once Method.

n_D	Constraints	Avg. ratio of equivalent Newton steps for Bbox (impl grad) over all-at-once method
2	none	21.8
10	none	18.9
10	1st Deriv.	2.2
10	1st and 2nd Deriv.	1.4

Table 3. Summary Comparison of Design Problem Formulations.

Problem Formulation	Robustness	Computational Cost	Independence of Optimization and Analysis Codes
Bbox (fd grad)	High	High	High
Bbox (impl grad)	High	Medium	Medium
all-at-once	Low	Low	Low

study, *smoothness plots* were obtained in the $n_D = 2$ design case. The smoothness plots show the change in the black-box objective function as one of the duct-defining spline coefficients is perturbed over a range between 0.9 and 1.1 of its nominal value. Figures 6-8, respectively, show the smoothness plots for the C^0, C^1 and C^∞ schemes. The continuity of the difference schemes is clearly reflected in the test results.

All the test results discussed thus far are for a fixed, coarse computational grid. Many analysis codes employ grid-refinement techniques to capture the details of flow features, such as shocks. Thus, it is of interest to consider the smoothness of the design problem when grid-refinement is used. Figure 9 shows the flow in the optimal duct for the C^∞ scheme when a simple technique is used to refine the grid near the shock. The shock detail is captured much more accurately using grid refinement than using the fixed, coarse grid.

The initial smoothness tests for the grid-refinement case were run using an objective function that is the sum of squares, on the coarse grid points, of the differences between the computed velocities and the goal velocities. Figure 10 is the smoothness plot for this objective function using the C^∞ scheme with grid refinement. (Similar results are obtained for the other difference schemes.) Considerable discontinuity is introduced into this objective function by grid refinement.

To remedy the discontinuities discussed above, a new *integral* objective function was used. For this objective, both the computed solution on the refined-grid and the velocity design goal were interpolated using cubic splines. The objective function was then defined as the numerical integral of the sum-of-squares of the differences between the two spline curves. Figure 11 shows the smoothness plot for the C^0 scheme with the new objective function and grid refinement. (Similar results are obtained for the other difference

schemes.) Figure 11 indicates that the combination of grid refinement and the integral objective function results in a smooth design problem, even for the C^0 scheme.

5. Conclusions

We have presented three methods for formulating design problems as optimization problems. The first is the black-box method where the optimization code is completely separated from the analysis code, and the optimization code repeatedly invokes the analysis code to provide values of the flow variables that are used to evaluate the objective function of the optimization. Most of these invocations of the analysis code are made by the optimization code in order to evaluate finite difference approximations to the gradients of the objective function (and constraints) with respect to the design variables. In general, this is very costly. We therefore presented a modification of the black-box method where these gradients are found by an algorithm based on the implicit function theorem. This black-box method with implicit gradients inherits most of the good properties of the black box-finite difference gradient method (good robustness, considerable independence of the optimization and analysis codes), while substantially reducing the computational cost.

The black-box method with implicit gradients can be retrofitted to most existing analysis codes to turn them into design codes. The amount of work required depends on the solution methodology employed in the analysis code. The largest task is to solve linear systems with a coefficient matrix that is the transpose of the Jacobian of the discretized flow equations with respect to the flow variables. In many cases (primarily in schemes based on Newton's method), this Jacobian is already computed by the analysis code. In other cases, it can readily be obtained by sparse differencing. In all cases, a solution method for the transpose of the Jacobian needs to be provided. While this is trivial if a direct factorization of the Jacobian is employed in the analysis code, such will rarely be the case for large scale (three dimensional) problems. It remains to be determined how iterative methods can best be adapted to solve transposed systems.

The other method we introduced was the all-at-once method where the optimization simultaneously varies the flow and design variables, and the discretized flow equations are viewed as equality constraints on the optimization. The primary difference between the all-at-once approach and the black-box approach is that the discrete flow equations are not required to be satisfied in the optimization iteration until the optimal solution is reached. Obviously, the optimization methodology and the flow equation solution methodology need to be tightly integrated in this approach, so that code independence is sacrificed. We found in our tests on the model duct flow problem that the all-at-once approach was less robust than the black-box approach, often failing to converge or converging to an undesirable local minimum. This was especially true when difference schemes of low continuity (those giving the sharpest shocks) were employed. However, when the all-at-once approach succeeded, it was dramatically less expensive than the other approaches. Since expense is a key issue for large problems, further investigation of how to increase the robustness of the all-at-once method seems justified.

The issue of smoothness of the black-box design problem was examined. It was concluded that the optimal design problem can be quite smooth, even in the presence of low continuity difference schemes and grid refinement, provided the design objective function is appropriately defined .

References

[1] J. W. Slooff, "A Survey of Computational Methods for Subsonic and Transonic Aerodynamic Design," in "Proceedings of International Conference on Inverse Design Concepts in Engineering Sciences" (G. S. Dulikravich, Ed.), pp. 1-67, University of Texas at Austin, 1984.

[2] G. R. Shubin, A. B. Stephens, and H. M. Glaz, "Steady Shock Tracking and Newton's Method Applied to One-Dimensional Duct Flow," Journal of Computational Physics, v. 39, n. 2, Feb. 1981, pp. 364-374.

[3] P. D. Frank and G. R. Shubin, "A Comparison of Optimization-Based Approaches for a Model Computational Aerodynamics Design Problem," Boeing Computer Services Technical Report ECA-TR-136, April, 1990.

[4] C. de Boor, A Practical Guide to Splines, Springer, New York, 1985.

[5] P. E. Gill, W. Murray, and M. H. Wright, Practical Optimization, Academic Press, New York, 1982.

[6] R. G. Voigt, "Requirements for Multidisciplinary Design of Aerospace Vehicles on High Performance Computers," ICASE Report No. 89-70, Sept. 1989.

[7] A. R. Curtis, M. J. D. Powell, and J. K. Reid, "On the Estimation of Sparse Jacobian Matrices," J. Inst. Math. Appl. 13, 1974, pp. 117-119.

[8] M. H. Rizk, "Aerodynamic Optimization by Simultaneously Updating Flow Variables and Design Parameters," AGARD Paper No. 15, May 1989.
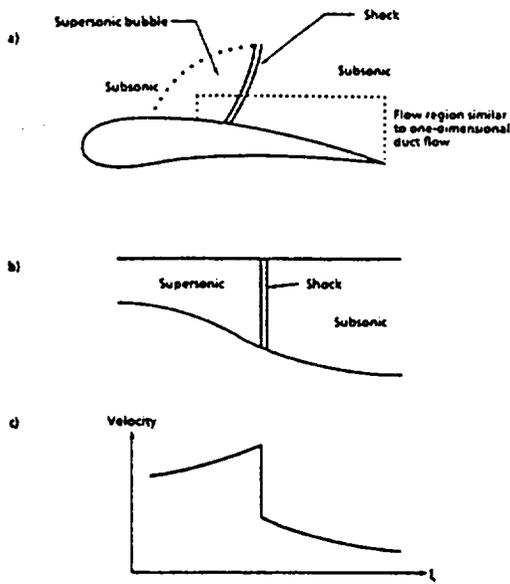
Figure 1: Analogy between a) flow over a transonic airfoil and b) flow in a duct. Of course, the flow in a) is two-dimensional, but is assumed one-dimensional in b). In either case, the velocity along the airfoil surface or along the duct length is qualitatively given in c).
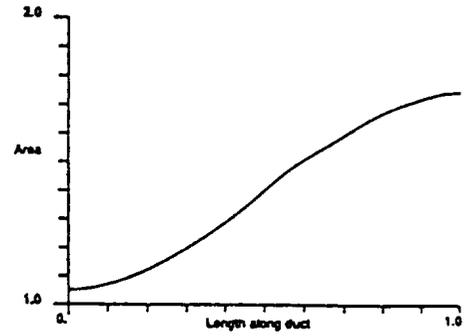
Figure 3a. Area profile, optimal duct with 10 design variables, no design constraints, using C^0 scheme
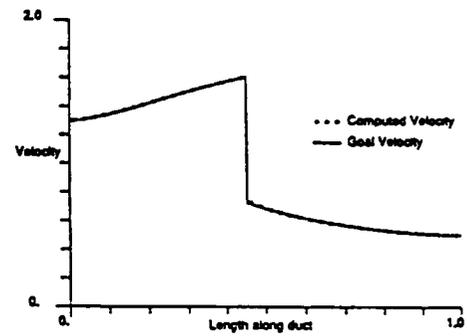
Figure 3b. Velocity profile, optimal duct with 10 design variables, no design constraints, using C^0 scheme
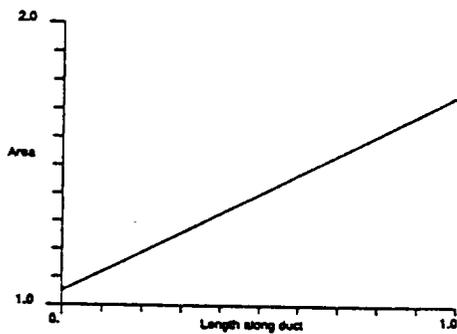
Figure 2a. Area profile, linear duct using C^0 scheme
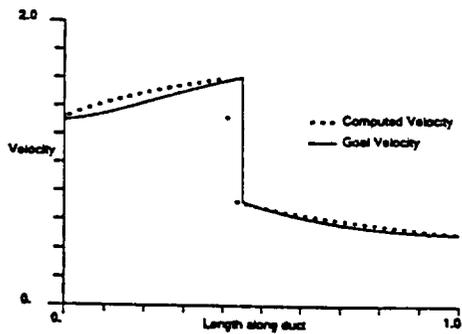
Figure 2b. Velocity profile, linear duct, using C^0 scheme
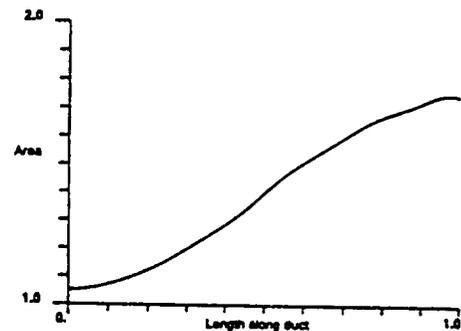
Figure 4a. Area profile, optimal duct with 10 design variables, no design constraints, using C^1 scheme
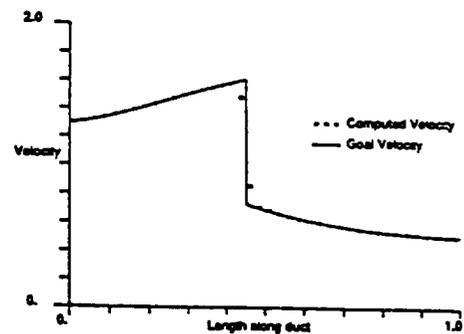
Figure 4b. Velocity profile, optimal duct with 10 design variables, no design constraints, using C^1 scheme

87
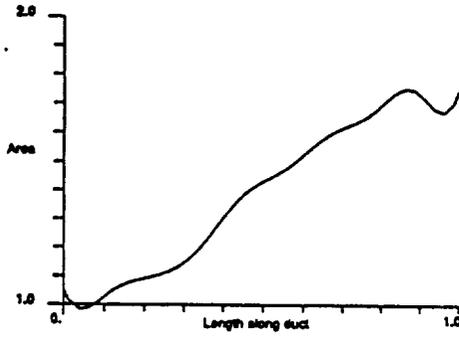
Figure 5a. Area profile, optimal duct with 10 design variables, no design constraints, using C∞ scheme
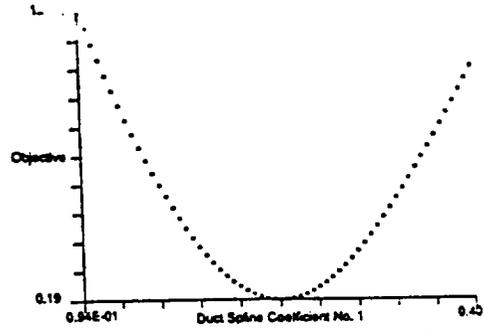
Figure 8. Objective as a function of duct geometry for the C∞ scheme
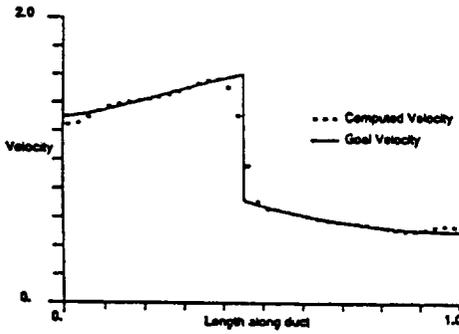
Figure 5b. Velocity profile, optimal duct with 10 design variables, no design constraints, using C∞ scheme
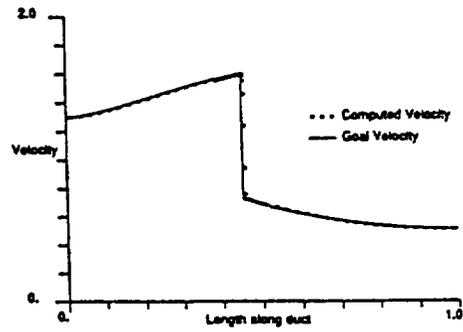
Figure 9. Velocity profile of optimal duct using grid refinement and the C∞ scheme
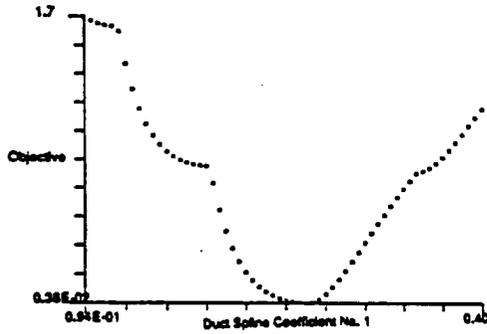
Figure 6. Objective as a function of duct geometry for the C⁰ scheme
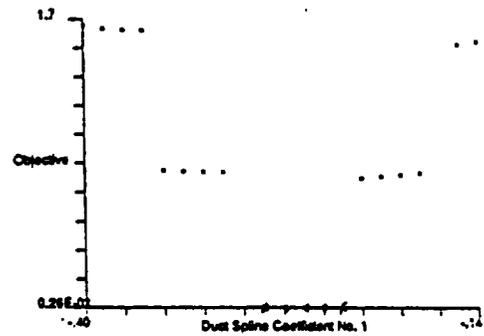
Figure 10. Objective as a function of duct geometry for the C∞ scheme with grid refinement
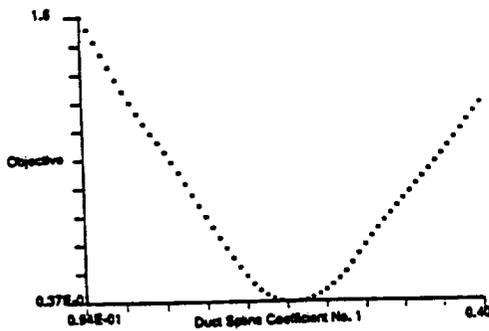
Figure 7. Objective as a function of duct geometry for the C¹ scheme
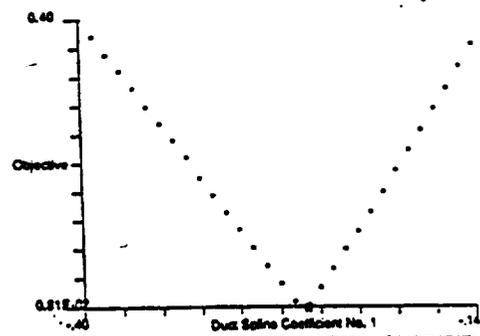
Figure 11. Integral objective as a function of duct geometry for the C⁰ scheme with grid refinement

DESIGN OF LAMINAR FLOW BODIES IN COMPRESSIBLE FLOW

Simha S. Dodbele, Research Scientist
Vigyan, Inc., Hampton, VA

ABSTRACT

An optimization method has been developed to design axisymmetric body shapes such as fuselages, nacelles and external fuel tanks with increased transition Reynolds numbers in subsonic compressible flow. The method involves a constraint minimization procedure coupled with analysis of the inviscid and viscous flow regions, and linear stability analysis of the compressible boundary-layer. Boundary-layer transition is predicted by a "hybrid" transition criterion based on Granville's transition criterion and a criterion using linear stability theory coupled with the e^n-method. A tiptank of a business-jet is used as an example to illustrate that the method can be utilized to design an axisymmetric body shape with extensive natural laminar flow. On the original tiptank boundary layer transition is predicted to occur at a transition Reynolds number of 6.04×10^6 on the original tiptank. On the designed body shape a transition Reynolds number of 7.22×10^6 is predicted using compressible linear stability theory coupled with e^n-method.

Introduction

Recent advances in airplane construction techniques and materials employing bonded and milled aluminum skins and composite materials allow for the production of aerodynamic surfaces without significant waviness and roughness, permitting long runs of natural laminar flow (NLF) over wings in subsonic flow. These advances lead to excellent opportunities for airplane drag reduction by increasing the extent of NLF over wings [1]. As compared to lifting surfaces laminar flow research on nonlifting air-frame surfaces, such as fuselages, nacelles, and external fuel tanks has received limited attention [2,3].

Reference 3 presents a recent overview of incompressible transition experiments on axisymmetric bodies. References 4-6 presented results of mostly, incompressible, under-water transition experiments over bodies of revolution with varying fineness ratio, indicating maximum transition-Reynolds numbers of about 20 million for low fineness ratio bodies.

A recent study [7] of bodies of revolution at high subsonic speeds without supersonic regions demonstrated the potential for tripling the length of sufficiently stable laminar flow at Mach number $(M) = 0.8$ and Length-Reynolds number $(R_L) = 40 \times 10^6$, in comparison with incompressible speed at the same length Reynolds number. A transition experiment was conducted in the NASA-Ames 12-ft. pressure tunnel by Boltz et al., [8,9] at high subsonic freestream Mach numbers, measuring the transition locations on two ellipsoids of fineness ratios (f_r) of 7.5 and 9.14. Transition occurred as far downsteam as 80 to 88% at $M = 0.90$ to 0.96. Reference 10 presents correlation of compressible boundary-layer-stability analysis done for several of the experimental results reported by Boltz et al. and indicates that integrated T-S linear logarithmic amplification factors (n-factors) of 8-11 are obtained at the point of measured transition onset.

The transition process over an axisymmetric body shape is caused by large amplitude growth of Tollmien-Schlichting (T-S) disturbance waves in the laminar boundary-layer flow. In compressible flow, the presence of density gradients in the boundary layer in the direction normal to the wall in addition to the velocity gradients can result in a large reduction in the spatial growth of T-S disturbances in the laminar boundary layer. The favorable damping effect of the T-S waves in compressible flow contribute to the achievement of increased transition-Reynolds numbers (R_{tr}) on lifting as well as

nonlifting aircraft surfaces in the absence of strong crossflow [7]. This favorable effect of compressibility should be exploited in the design of advanced NLF bodies for application to general aviation, commuter, transport and business aircraft.

This paper presents a new design method in which Granville's transition criterion and the e^n-method (originally introduced by Smith [11] and Van Ingen [12]) has been incorporated to generate body shapes with increased transition Reynolds numbers at subsonic compressible speeds. Design calculations for a tiptank in compressible flow are presented as an example case.

Optimization Procedure for NLF Body Design

The design method developed to obtain body shapes with extensive runs of laminar flow is illustrated in the flowchart (Fig. 1). Initial values of the design variables describing the body shape are input along with the length Reynolds number, Mach number of the free stream, and the fineness ratio of the desired body shape. The axisymmetric body is described by design variables representing the body ordinates in the forebody section and in the afterbody section

A constrained minimization method (CONMIN) [13] is coupled with analysis of the inviscid and viscous flow regions, linear stability analysis of the compressible boundary-layer and a transition prediction method. The aerodynamic analysis program used in the present optimization procedure is based on a low-order surface-singularity method (VSAERO) [14]. Pressure distributions and velocity distributions can be computed by this method which uses surface singularity panels to represent the body shape. The boundary-layer profiles along the surface of the body, required for the e^n-method, are generated by a modified axisymmetric boundary layer code (HARRIS) [15]. The boundary-layer finite difference program calculates detailed boundary-layer velocity and temperature profiles along with their first and second derivatives normal to the surface, including the effects of transverse curvature. Analysis of the laminar boundary-layer stability along the body is done by using compressible linear stability theory. The COSAL program [16] solves the finite-differenced, boundary-layer stability equations by using matrix methods. The compressible T-S eigenvalue problem is solved for each boundary-layer station along the body surface giving temporal growth rates of the instability waves propagating at specific wavelengths and wave angles. The temporal growth rates are transformed to the spatial growth rates using Gaster's phase-velocity relationship [17]. Boundary-layer transition is predicted by the e^n-method in which n, usually referred to as n-factor, is obtained by integrating the linear growth rate of the T-S waves from the neutral stability point to a location downstream of the body.

The correlation of a large number of wind tunnel data and flight transition experiments with linear boundary-layer stability calculations has made the e^n-method a consistent transition-prediction method [18]. For experiments in wind tunnels with low turbulence and low acoustic levels the onset of transition can be correlated with an n-factor of 9 to 11 in subsonic, transonic and supersonic flows. In the case of flight tests, higher n-factors of the order of 12 to 15 have been observed to correlate transition. In the present design calculations, the n-factor in the design method can be chosen so as to suit a particular application- e.g., to design a body for a wind tunnel, a flight test article, or an under-water body.

A number of geometric and aerodynamic constraints are imposed on the design parameters to generate practical and realistic body shapes for given design conditions. The geometric constraints will be that the design variables are constrained by the specified upper and lower bounds. Judicious choice of the upper and the lower bounds for the design variables will accelerate convergence of the solutions. The level and the location of the minimum surface pressure along the body surface are aerodynamically constrained by the requirement that the turbulent boundary layer over the aft-portion of the body should not separate until x=0.95 for the design conditions. The objective function is taken to be a function of the location of transition predicted by the following "hybrid" transition criterion.

90

$$f_{obj}=1- x_{tr}(h) \qquad \text{where } x_{tr}(h) = (x_{tr}(g)+ x_{tr}(e^n))/2, \qquad (1)$$

$x_{tr}(g)$ is the transition location predicted by using Granville's transition criterion and $x_{tr}(e^n)$ is the transition location predicted by using e^n-method with an n-factor of 9. But for calculating the gradients of the objective function Granville's transition criterion is used for predicting transition.

The objective function given by Eqn. (1) is to be minimized subject to the constraints on the design variables. The optimizer computes gradients of the objective function using Granville's transition criterion and then, using either a conjugate direction method or a method of feasible direction, determines a linear search direction, along which a new constrained variable is constructed.

An improved or minimum feasible objective functional value is calculated by using the hybrid transition criterion given by Eqn. (1) and a series of proposed updated design variables are calculated. The objective function and the constrained function are evaluated using the updated design variables, interpolating over the range of feasible proposed design variables resulting in a minimum value of the objective function. The results are tested against a convergence criteria. The procedure will stop if the convergence criterion is satisfied, giving a body shape with maximum transition length satisfying the separation constraint. If the convergence criterion is not satisfied the design parameters go through the analyzer again resulting in a new set of design variables and the procedure is repeated until a final body shape is obtained.

Details of the Design Method

The present computational procedure is used to design axisymmetric bodies at zero incidence. At zero incidence, the growth of the two-dimensional, T-S disturbances is the most dominant instability mechanism on an axisymmetric body leading to transition in the boundary layer if laminar separation does not happen earlier than natural transition. For the aerodynamic analysis, the body is modelled by 32 panels in the axial direction and 8 panels in the circumferential direction. Using the VSAERO panel method inviscid pressure distributions were obtained and interpolated at 200 axial stations. The boundary-layer velocity and temperature profiles are obtained with 101 points in the direction normal to the surface and 90 stations in the streamwise direction. Presently, in the design method the boundary layer calculations are carried out for adiabatic wall conditions and zero suction through the wall.

The boundary-layer stability equations for the example considered are solved at every 5th streamwise boundary-layer station starting from the first station. The boundary-layer stations are skipped from the point of view of reducing the computational time. In the global search for eigen values, the sixth-order stability equation is solved at each chordwise station and in the local search for the eigen values the full eighth order stability equation is solved. Prior knowledge of the critical boundary-layer disturbance frequencies, which are functions of the Mach number helps to identify the critical frequency spectrum during the course of the design optimization. To assess the effect of extending the length of laminar boundary-layer flow over the geometries analyzed, calculation of the viscous drag is made using a modified integral boundary layer approach [3].

Computational Results and Discussions

To increase the speed of computations the design program is run with the initial body geometry using Granville's transition criterion to obtain a converged body shape. The final body shape obtained by using Granville's criterion is then used as the initial geometry input into the design program and the hybrid transition criterion is selected. This procedure greatly reduces computer time and also results in rapid convergence of the design variables.

91

Results obtained by the optimization procedure are discussed through an example. All the computations were done on a CRAY-2 computer. A body of revolution whose maximum diameter and length correspond to those of a tiptank of a representative business aircraft is considered. The tiptank has a fineness ratio of 8.00 and the design flight conditions considered for the present calculations are given by M=0.7 and unit Reynolds number (R') =1.28 x 10^6/foot. The axisymmetric body is modelled by a set of 27 body coordinates with 12 points defining the forebody section and 15 points defining the aftbody section. Twelve design variables representing the ordinates in the forebody region are allowed to vary within the set of specified upper and lower bounds while simultaneously holding the tail section aft of the maximum thickness point unchanged during the design iterations.

The final body shape obtained at the end of the optimization with Granville's criterion is used as input data to the design program with the hybrid transition criterion. In the present example, since the axisymmteric flow is subcritical zero TS wave angle is assumed in the design calculations.

The original tiptank and the final body shape obtained using the hybrid transition criterion along with the results of stability analyses are shown in Fig. 2 for comparison. The envelope for the new body shape has a smaller gradient than on the original tiptank shape. On the original body, the critical disturbance characterized by a frequency of 3500 Hz starts growing after 13% of the body length from the nose and reaches an n-factor of 9 at $x_{tr}(e^n) = 0.33$ $(R_{tr}(e^n) = 6.04 \times 10^6)$. A drag coefficient (C_D) of 0.0491 is predicted on the original tiptank with the boundary-layer transition fixed at x_{tr}=0.33. The design program took 2785 secs. to predict the final design shape with the hybrid transition criterion.

On the designed body shape the transition location corresponding to n-factor of 9 occurs at $x_{tr}(e^n) = 0.39$ $(R_{tr}(e^n) = 7.22 \times 10^6)$. Though the critical frequency leading to transition remains at 3500 Hz on the original tiptank and the designed body, boundary-layer transition as predicted by the en-method occurs much further downstream on the designed body. A drag coefficient (C_D) of 0.0415 is predicted on the designed body shape with the boundary-layer transition fixed at x_{tr}=0.39

Conclusions

An optimization procedure has been developed to design axisymmetric body shapes with increased transition Reynolds number. The new design method involves a constraint minimization procedure coupled with analysis of the inviscid and viscous flow regions, and linear stability analysis of the compressible boundary-layer. Boundary-layer transition is predicted by a "hybrid" transition criterion based on Granville's transition criterion and a criterion based on linear stability theory combined with the e^n-method. A tiptank of a business-jet is given as an example to demonstrate that the method can be used to design an axisymmetric body shape with increased transition Reynolds number. Boundary-layer transition is predicted to occur at a transition Reynolds number of 6.04 x 10^6 on the original tiptank. On the designed body shape a transition Reynolds number of 7.22 x 10^6 is predicted using the en-method, an increase of 20% in transition Reynolds number.

Acknowledgments

References

[1] Holmes, B.J., Obara, C.J., and, Yip, L, "Natural Laminar Flow Experiments on Modern Airplane Surfaces," NASA TP-2256, June 1984.

[2]   Dodbele, S.S., Van Dam, C.P., and Vijgen, P.M.H.W., "Design of Fuselage Shapes for Natural Laminar Flow," NASA CR-3970, March 1986.

[3]   Dodbele, S.S., Van Dam, C.P., and Vijgen, P.M.H.W., Holmes, B.J., "Shaping of Airplane Fuselages for Minimum Drag," Journal of Aircraft, Vol. 24, May 1987, pp. 298-304.

[4]   Carmichael, B.H., "Underwater Vehicle Drag Reduction through Choice of Shape," AIAA paper no. 66-657, 1966.

[5]   Carmichael, B.H., "Computer Study to Establish the Lower Limit of Length-to-Diameter Rates Advisable for Low Drag Bodies," SID64-1938, North American Aviation, Inc., 1964

[6]   Carmichael, B.H., "Underwater Drag Reduction Through Optimal Shape," In Underwater Missile Propulsion, edited by L. Greiner, Compass Publications, Inc., Arlington, VA 1966.

[7]   Vijgen, P.M.H.W., Dodbele, S.S., Holmes, B.J., Van Dam, C.P., "Effects of Compressibility on Design of Subsonic Fuselages for Natural Laminar Flow," Journal of Aircraft, Vol. 25, Sept. 1988, pp. 776-782.

[8]   Boltz, E.W., Kenyon, G.C., and Allen, C.Q., "The Boundary-Layer Transition Characteristics of Two Bodies of Revolution, a Flat Plate, and an Unswept Wing in a Low-Turbulence Wind Tunnel," NASA TN D-309, April 1960.

[9]   Boltz, E.W., Kenyon, G.C., and Allen, C.Q., "Measurements of Boundary-Layer Transition at Low Speeds on Two Bodies of Revolution in a Low-Turbulence Wind Tunnel," NACA Research Memorandum, RM A56G17, Sept 1956.

[10] Vijgen, P.M.H.W., Dodbele, S.S., Pfenninger, W., Holmes, B.J., "Analysis of Wind-Tunnel Boundary-Layer Transition Experiments on Axisymmetric Bodies at Transonic Speeds Using Compressible Boundary Layer Stability Theory," AIAA paper no. 88-0008, 1988, to appear in AIAA Journal.

[11] Smith, A.M.O., "Transition, Pressure Gradient and Stability Theory," IX International Congress for Applied Mechanics, Brussels, 1956.

[12] Van Ingen, J.L., " A Suggested Semi-Empirical Method for the Calculation of the Boundary Layer Transition Region," University of Technology, Department of Aerospace Engineering, VTH-74, Delft, 1956.

[13] Vanderplaats, G. N, "CONMIN - A Fortran Program for Constrained Function Minimization," NASA TMX- 62282, 1973.

[14] Maskew, B., "Prediction of Subsonic Aerodynamic Characteristics- A Case for Low-Order Panel Methods," Journal of Aircraft, Vol. 19, No. 2, Feb. 1982, pp. 157-163.

[15] Harris, J.E., and Blanchard, D.K., "Computer Program for Solving Laminar, Transitional, or Turbulent Compressible Boundary-Layer Equations for Two-Dimensional and Axisymmetric Flow," NASA TM 83207, Feb 1982.

[16] Malik, M.R., "COSAL- A Black-Box Compressibility Stability Analysis Code for Transition Prediction in Three-Dimensional Boundary Layers," NASA CR 165925, May 1982.

[17] Gaster, M., "Propagation of Linear Wave Packets in Laminar Boundary-Layers," AIAA Journal, Vol. 19, 1981, pp. 419-423.

[18] Bushnell, D.M., Malik, M.R., and Harvey, W.D., "Transition Prediction in External Flows via Linear Stability Theory," IUTAM Symposium Transonicum III, Gottingen, West Germany, May 24-27, 1988.
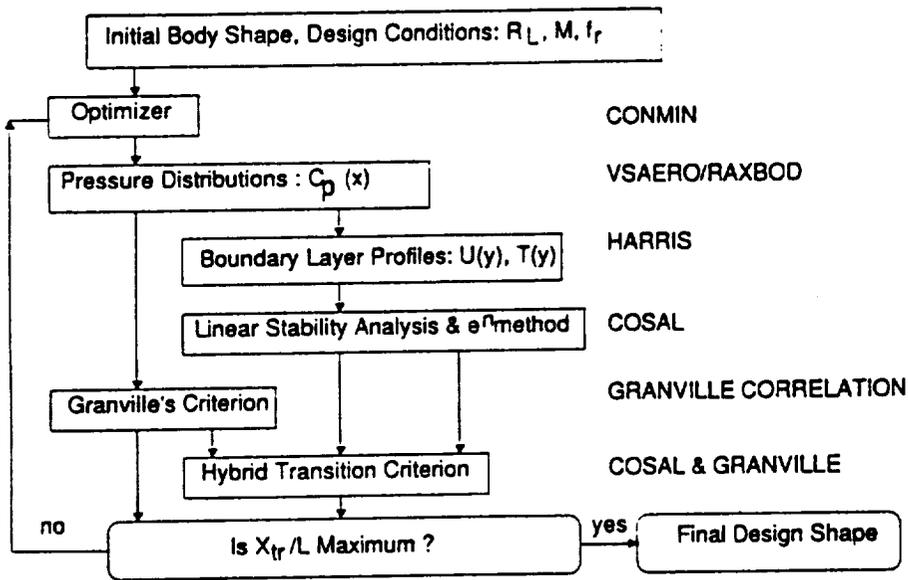
Fig. 1. Flow Chart of design procedure for NLF fuselage
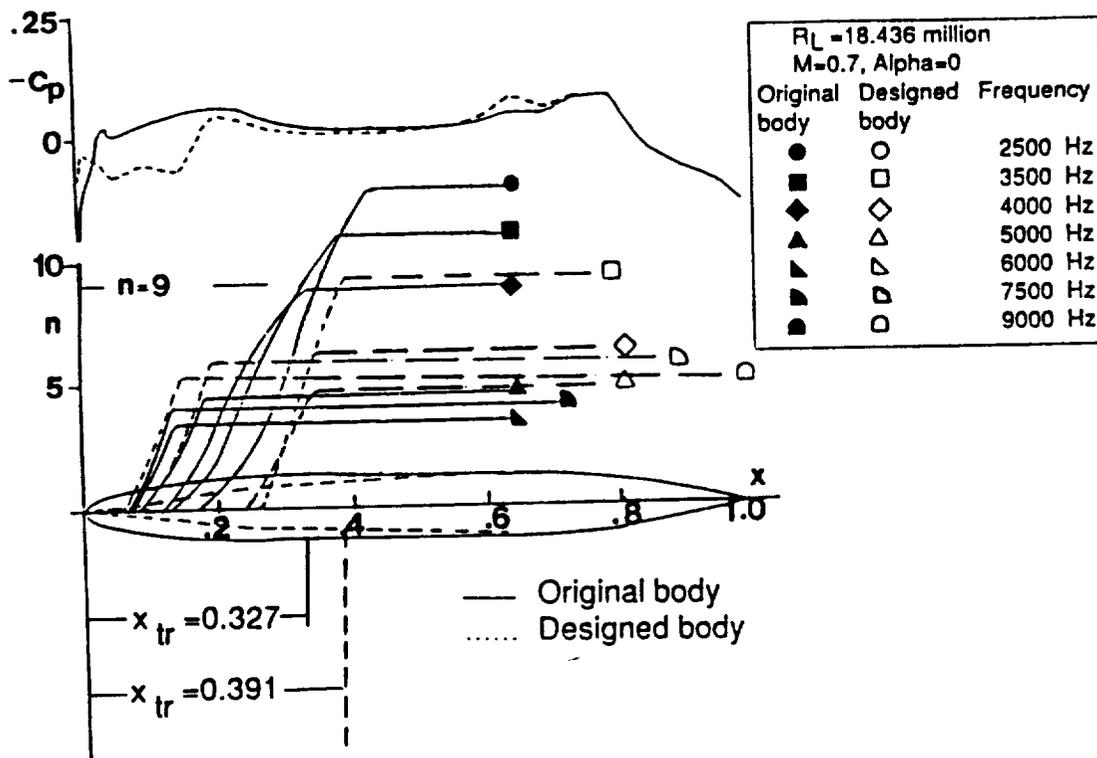
Fig. 2. Pressure distributions, predicted compressible T-S disturbance growth curves and transition locations on the original tiptank and the final design shape (optimization by hybrid transition criterion)

IMPROVED APPROXIMATIONS FOR DYNAMIC DISPLACEMENTS
USING INTERMEDIATE RESPONSE QUANTITIES*

Harold L. Thomas, *Abdon E. Sepulveda*, and *Lucien A. Schmit*
University of California, Los Angeles, California
4531 Boelter Hall, UCLA, Los Angeles, CA 90024-1593

Abstract

An approximation for dynamic displacements, which captures the nonlinearities associated with resonance, is presented. This approximation is constructed using approximate intermediate response quantities. When dynamic displacements are constrained using this high quality approximation, frequency constraints are no longer needed to keep the design away from resonance.

Introduction

During the structural synthesis process designs that have natural frequencies near the forcing frequencies of the applied loads may be generated. This produces a resonance condition with large dynamic displacements that are very nonlinear functions of changes in the design variables. The usual approach to this problem is to place frequency constraints on the design to keep it away from the loading frequencies. Choosing the values of these frequency constraints is difficult because if they are too close to the loading frequencies, near resonance will occur and if they are too far away the design may be overly conservative. In this work an approximation for dynamic displacements is developed which captures the nonlinear effects of resonance. The approximations introduced here allow the designer to forego the use of difficult to select frequency constraints. An important feature of the approximations presented is that they can be used in the context of modal analysis for dynamic structural response.

Approximation Concepts

The use of the approximation concepts method is needed for efficient structural synthesis. In this method an explicit approximate optimization problem is formulated and solved at each design stage. In the mid 1970's approximate representations for constraints and objective functions were generated using first order Taylor series expansions in terms of direct or reciprocal sizing types design variables (see Refs. 1 and 2). More accurate approximations can be constructed using approximations of intermediate response quantities, which were introduced in Ref. 2. The intermediate response quantity idea has been applied to stress constraints (Ref. 3), frequency constraints (Ref. 4), and steady state harmonic displacement and force constraints as well as complex eigenvalues constraints (Ref. 5). In this approach simple approximations (e.g. linear reciprocal, hybrid) of intermediate response quantities (e.g. forces in the case of stress constraints and modal energies in the case of frequency constraints) in terms of design variables can be used while retaining the explicit nonlinear dependence of the constraints on the intermediate response quantities.

* Graduate Research Assistant, currently Research Engineer, VMA Engineering, Goleta, CA.
** Postdoctoral Research Associate.
*** Professor of Engineering and Applied Science.

Mathematically:

$$\bar{R}_I = f(X) \tag{1}$$

where the approximate intermediate response ($\bar{R}_I$) is a simple function of the design variables (X). The approximate value of the constrained response ($\bar{R}_C$) is then calculated as:

$$\bar{R}_C = g(\bar{R}_I, X) \tag{2}$$

In other words, given the design vector X, the approximate value(s) of the intermediate response ($\bar{R}_I$) is calculated first using Eq. 1. Then the constrained response $\bar{R}_C$ corresponding to $\bar{R}_I$ and X is evaluated using Eq. 2.

Approximations for Dynamic Displacements

The matrix equation of motion for an undamped structure is:

$$[M]\{\ddot{u}\} + [K]\{u\} = \{P\} \tag{3}$$

Assuming a sinusoidal loading and response at frequency Ω:

$$\{P\} = \{p\}\sin\Omega t \tag{4}$$

$$\{u\} = \{a\}\sin\Omega t \tag{5}$$

Equation (3) can be transformed into the frequency domain:

$$(-\Omega^2[M] + [K])\{a\} = \{p\} \tag{6}$$

In the usual approach to approximating the dynamic displacements, Eq. (6) is solved directly for $\{a\}$ and the derivatives of the a_i are found by implicitly differentiating Eq. (6):

$$\frac{\partial}{\partial x_j}\{a\} = [-\Omega^2 M + K]^{-1}\left(\frac{\partial}{\partial x_i}\{p\} - \left[-\Omega^2\frac{\partial M}{\partial x_i} + \frac{\partial K}{\partial x_i}\right]\{a\}\right) \tag{7}$$

The dynamic displacements are then approximated as direct, reciprocal, or hybrid (see Ref. 6) functions of the design variables so that

$$\ddot{u}_i = \bar{a}_i\sin\Omega t = f(X)\sin\Omega t \tag{8}$$

The hybrid approximation uses either a direct or reciprocal expansion in each of the design variables, selecting term by term the alternative that is most conservative. It is commonly used for static and dynamic displacement constraints and it will be employed here to construct the full order solution for comparison purposes.

The approximation shown in Eq. (8) has two drawbacks. The first is that it is based on the direct solution of Eqs. (6) and (7) which is very expensive for large problems. The second drawback is that it is a poor approximation for the response whenever the loading frequency (Ω) is near a natural frequency of the structure, because of the strong nonlinear effects of resonance.

96

In order to determine the dynamic response of large structures, modal analysis is often used to reduce the order of Eq. (6). In modal analysis the response of the structure is approximated as a linear sum of an. orthogonal set of basis vector $\{\phi_n\}$, that is

$$\{a\} = \sum_{n=1}^{N} \{\phi_n\} z_n = [\Phi] \{z\} \tag{9}$$

where the z_n are called the modal participation coefficients. The first N natural vibration modes of the structure (eigenvectors) are usually chosen as the basis vectors $\{\phi_n\}$ and that is the approach used in this work. Substituting Eq. (9) into Eq. (6) and pre-multiplying by $[\Phi]^T$ gives

$$[\Phi]^T (-\Omega^2[M] + [K]) [\Phi] \{z\} = \{f\} \tag{10}$$

where

$$\{f\} = [\Phi]^T \{p\} \tag{11}$$

Defining the modal potential energy matrix

$$[U] = [\Phi]^T [K] [\Phi] \tag{12}$$

and the matrix

$$[T] = [\Phi]^T [M] [\Phi] \tag{13}$$

makes it possible to rewrite Eq. (10) in the following form

$$(-\Omega^2 [T] + [U]) \{z\} = \{f\} \tag{14}$$

Note that the order of Eq. (14) is N which is much smaller than the order of the full system. The value of N is chosen to be the number of modes needed to accurately represent the structural response.

When the basis vectors $\{\phi_n\}$ are the natural mode shapes of structure they are orthogonal to both $[K]$ and $[M]$ so that $[U]$ and $[T]$ are diagonal matrices. Equation (14) is then decoupled and the individual z_n are calculated using the following expression

$$z_n = \frac{f_n}{U_n - \Omega^2 T_n} \tag{15}$$

The approximation introduced in this work is constructed as follows. Evaluate the derivatives of U_n and T_n with respect to the design variables (X) assuming the eigenvectors are invariant, so that:

$$\frac{\partial U_n}{\partial x_j} = \{\phi_n\}^T \frac{\partial [K]}{\partial x_j} \{\phi_n\} \tag{16a}$$

and

$$\frac{\partial T_n}{\partial x_j} = \{\phi_n\}^T \frac{\partial [M]}{\partial x_j} \{\phi_n\} . \tag{16b}$$

Approximate U_n and T_n as linear functions of the design variables (see Refs. 4 and 5):

$$\bar{U}_n = U_{n_0} + \sum_{j=1}^{NDV} \frac{\partial U_n}{\partial x_j}(x_j - x_{j_0}) \tag{17a}$$

$$\bar{T}_n = T_{n_0} + \sum_{j=1}^{NDV} \frac{\partial T_n}{\partial x_j}(x_j - x_{j_0}) \tag{17b}$$

where NDV is the number of design variables. Calculate the approximate modal participation coefficients as

$$\bar{z}_n = \frac{f_n}{\bar{U}_n - \Omega^2 \bar{T}_n} \tag{18}$$

and finally the approximate amplitudes of displacement as

$$\{\bar{a}\} = [\Phi]\{\bar{z}\} \tag{19}$$

The error associated with this approximation is small and comes from two assumptions. The first is that the displacements can be represented by a truncated set of modes (N). This is the error associated with the analysis and it can be controlled by choosing a satisfactory value for N. The other assumption is that the mode shapes are invariant with changes in the design variables. This error can be controlled by putting move limits (M.L.) on the design variables at each design stage. In the example section of this paper it is shown that 60% move limits are not unreasonable.

The reason for the accuracy of the approximation near resonance is now examined. If the numerator and denominator of the right hand side of Eq. (18) are divided by $\bar{T}_n$ the result is

$$\bar{z}_n = \frac{f_n/\bar{T}_n}{\bar{U}_n/\bar{T}_n - \Omega^2} \tag{20}$$

Note that $\bar{U}_n/\bar{T}_n$ is the Rayleigh Quotient Approximation (see Ref. 4) for the structural eigenvalue ($\bar{\lambda}_n$) corresponding to n^{th} natural mode. Therefore, Eq. (20) can be rewritten as

$$\bar{z}_n = \frac{f_n/\bar{T}_n}{\bar{\lambda}_n - \Omega^2} \tag{21}$$

Note that as the structural eigenvalue for mode n ($\bar{\lambda}_n$) approaches the loading frequency (Ω), the modal participation coefficient for the nth mode ($\bar{z}_n$) becomes very large, which is exactly what happens at resonance. The full order approximation in Eq. (8) cannot capture this effect.

It should be recognized that an approximation based on the direct solution of Eq. (6) which does capture the effect of resonance was presented in Ref. 7. However, this approximation cannot be employed when modal analysis is used to solve for the dynamic response of the structure.

Examples

The example used in this paper will be the mass minimization of the antenna structure shown in Fig. 1. The structure is modeled with 10 beam type finite elements has 24 degrees of freedoms, and five modes are used for the modal analysis. The elements are linked to produce a symmetric structure with five design elements (see Fig. 1). Each design element cross section (see Fig. 2) has 2 design variables (t_h and t_b) for

a total of 10 design variables. The response of node 7 in the y-direction due to a 500 N load applied in the y-direction at node 9 with various forcing frequencies is shown by the solid line in Fig. 3. The off center load will excite both bending and torsion in the structure. Note the response peaks near the first natural frequency of 0.43 Hz (first bending mode) and the second natural frequency of 1.04 Hz (first torsional mode).

In the first example problem the structure is loaded at 0.7 Hz, which is away from the resonance peaks. The displacement amplitudes of nodes 5 and 7 in the y-direction are constrained to be less than 1 cm. There are no other constraints on the structure. The design histories and error in the displacement amplitude approximations are shown in Table 1 for both the modal and direct approximations for 30% and 60% design variable move limits. The design histories are plotted in Fig. 4. Note that with 30% move limits both approximations perform reasonably well with 60% move limits the modal approximation gives much faster design convergence. Also note that the modal approximation is more accurate and is still accurate when 60% move limits are used. The response of the final design at various loading frequencies is shown by the solid line in Fig. 3. In the final design the constraint on node 5 is active. The final design is presented in Table 2.

The second example problem is the same as the first except the loading frequency is now 0.5 Hz, which is near resonance, and displacement amplitudes of nodes 5 and 7 in the y-direction are constrained to be less than 10 cm. The design histories and error in the approximations are shown in Table 3. The design histories are plotted in Fig. 5. Note that with 30% move limits the modal approximation converges rapidly while the direct approximation overshoots the optimum and oscillates with designs that have about 7% infeasibility with respect to the displacement amplitude constraints. Even when 60% move limits are used, the modal approximation is quite accurate and a near final design is achieved after only 4 iterations. At the final design the constraint on node 5 is active. The final design is shown in Table 2.

Conclusions and Recommendations

The modal approximation for dynamic displacement response is quite accurate even when the design is near resonance and large move limits are used.

The approximation presented in this work can be extended to damped response problems when modal analysis is used. In these problems the quantities u_i, a_i, z_n, $\{\phi_n\}$, U_n and T_n are all complex but the development of the approximation is quite similar. Transient response of damped structures can also be approximated in the manner presented in this work.

Finally, the problem of disjoint design spaces that occurs when constraints are placed on dynamic displacements, observed in Ref. 8 and explained in Ref. 9, can be attacked using this kind of approximation. By temporarily setting a particular modes participation coefficient equal to zero the design will be able to "jump across" the resonance peak associated with this mode. This may help the design process converge to a global optimum.

Acknowledgement

References

1. Schmit, L.A. and Farshi, B., "Some Approximation Concepts for Efficient Structural Synthesis," *AIAA Journal*, Vol. 12, May 1974, pp. 692-699.

2. Schmit. L.A. and Miura, H., "Approximation Concepts for Efficient Structural Synthesis," NASA CR 2552, March 1976.

3. Vanderplaats, G.N. and Salajegheh, E., "A New Approximation Method for Stress Constraints in Structural Synthesis," *AIAA Journal*, Vol. 27, March 1989, pp. 352-358.

4. Canfield, R.A., "An Approximation Function for Frequency Constrained Structural Optimization," *Proceedings of the Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization*, AIAA, Hampton, VA, Sept. 28-30, 1988, pp. 937-953.

5. Thomas, H.L. and Schmit, L.A., "Improved Approximations for Control Augmented Structural Synthesis," *Proceedings of the AIAA/ASME/ASCE/AHS/ASC 31st Structures, Structural dynamics and Materials Conf.*, AIAA, New York, April 1990, pp. 227-294.

6. Starnes, J.R., Jr. and Haftka, R.T., "Preliminary Design of Composite Wings for Buckling, Stress and Displacement Constraints," *Journal of Aircraft*, Vol. 16, Aug. 1979, pp. 564-570.

7. Miura, H. and Cnargin, M., "Automated Tuning of Airframe Vibration by Structural Optimization," *Proceedings of the 42nd Annual Forum and Display American Helicopter Society*, Washington, D.C., June 2-4, 1986.

8. Cassis, J.H., "Optimum Design of Structures Subjected to Dynamic Loads," UCLA-ENG-7451, June 1974, UCLA, School of Eng. and Applied Sci., Los Angeles, Calif.

9. Johnson, E.H., "Disjoint Design Spaces in the Optimization of Harmonically Excited Structures," *AIAA J.*, Vol. 14, Feb. 1976, pp. 259-261.

Table 1

Design Histories for Example 1

Design Stage	Mass (kg) [% Error in Approximation]			
	Modal (Intermediate Response) 30% M.L.	Full Order (Hybrid) 30% M.L.	Modal (Intermediate Response) 60% M.L.	Full Order (Hybrid) 60% M.L.
0	4982	4982	4982	4982
1	5991 [0.0]	5991 [0.6]	5881 [0.2]	5708 [4.5]
2	5903 [0.3]	5792 [2.8]	5321 [0.3]	5560 [2.0]
3	5562 [0.3]	5618 [2.2]	5299 [0.0]	5417 [0.1]
4	5436 [0.1]	5497 [0.6]	5296 [0.0]	5351 [0.2]
5	5368 [0.1]	5424 [0.6]	5289 [0.1]	5324 [0.1]
6	5340 [0.1]	5360 [0.2]		5308 [0.1]
7	5313 [0.0]	5329 [0.2]		5301 [0.0]
8	5302 [0.0]	5308 [0.0]		5299 [0.0]
9	5299 [0.1]	5302 [0.0]		
10	5296 [0.2]	5299 [0.0]		

Table 2

Final Designs

Design Element	Design Variable	Example 1	Example 2
1	t_h	10.00[a]	0.52
	t_b	10.00[a]	0.77
2	t_h	10.00[a]	0.50[b]
	t_b	10.00[a]	0.50[b]
3	t_h	1.46	0.50[b]
	t_b	0.50[b]	0.50[b]
4	t_h	10.00[a]	2.59
	t_b	10.00[a]	2.68
5	t_h	3.59	0.50[b]
	t_b	3.92	0.50[b]

[a] At upper bound
[b] At lower bound

Table 3

Design Histories for Example 2

Design Stage	Mass (kg) [% Error in Approximation]		
	Modal (Intermediate Response) 30% M.L.	Full Order (Hybrid) 30% M.L.	Modal (Intermediate Response) 60% M.L.
0	4982	4982	4982
1	3749 [0.0]	3749 [3.4]	2508 [0.4]
2	2898 [0.3]	2995 [17.4]	1368 [0.6]
3	2245 [0.2]	2344 [15.3]	1180 [0.3]
4	1748 [0.2]	1835 [12.0]	1175 [0.0]
5	1401 [0.3]	1472 [2.1]	1171 [0.1]
6	1219 [0.2]	1264 [2.6]	1170 [0.0]
7	1129 [0.1]	1181 [1.6]	1169 [0.0]
8	1169 [0.4]	1163 [3.0]	
9	1168 [0.0]	1155 [5.6]	
10	1165 [0.1]	1137 [6.4]	
11		1146 [5.6]	
12		1137 [6.0]	
13		1140 [7.3]	
14		1125 [7.7]	
15		1140 [6.9]	
16		1127 [7.6]	
17		1138 [7.9]	
18		1123 [8.3]	
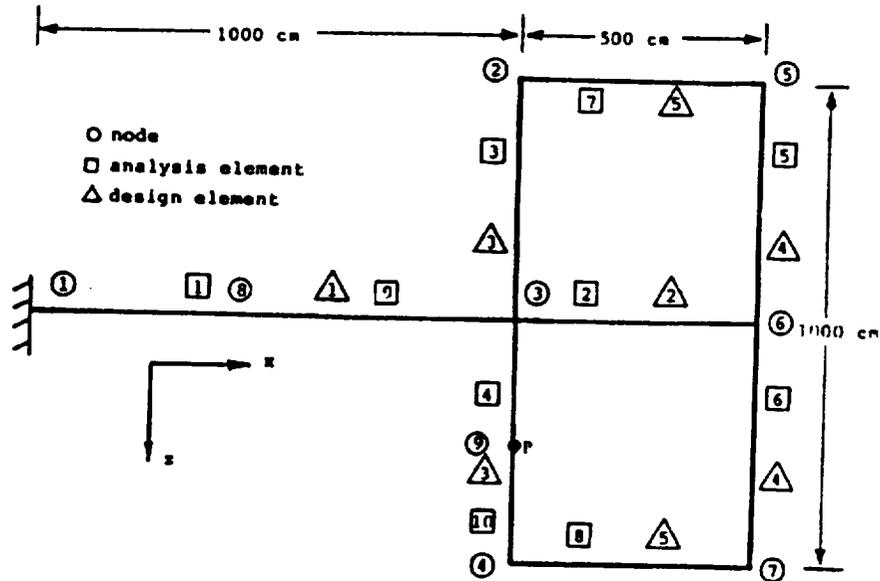
Figure 1. Antenna Structure

$$B = 7.10 \times 10^6 \text{ Nt/cm}^2$$
$$\rho = 2.788 \times 10^{-3} \text{ kg/cm}^3$$
$$\nu = .3$$
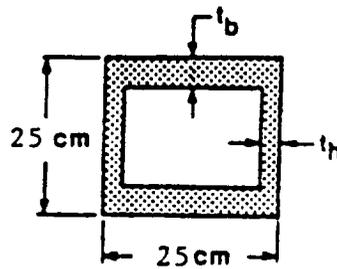
Figure 2. Beam Element Cross Section
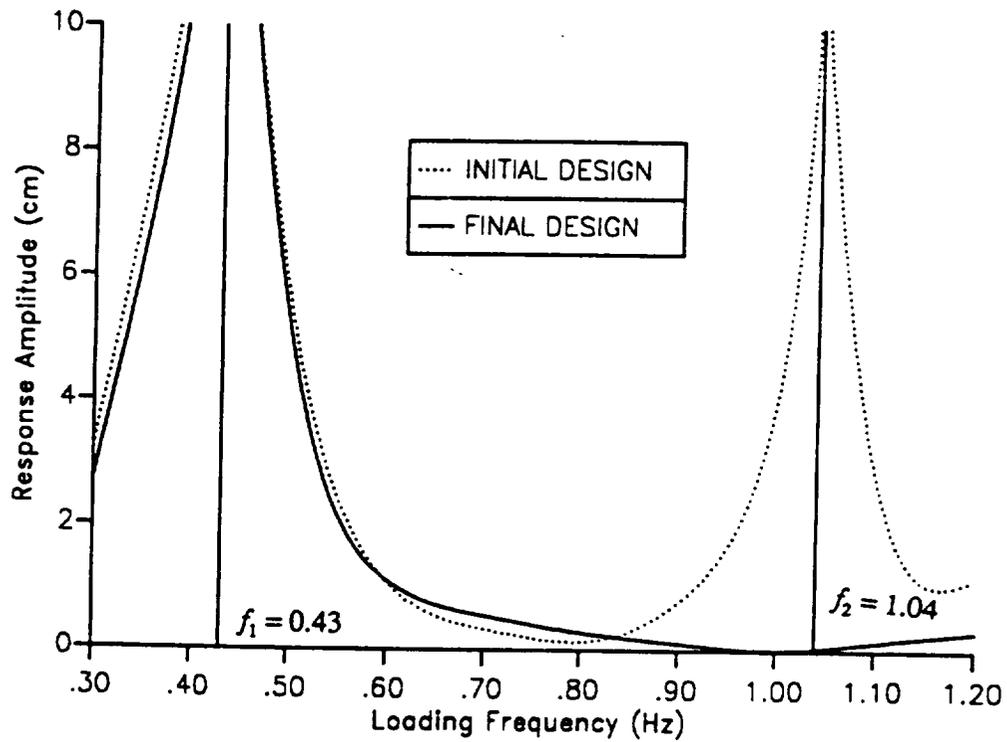
Figure 3. Structural Response at Various Frequencies

103
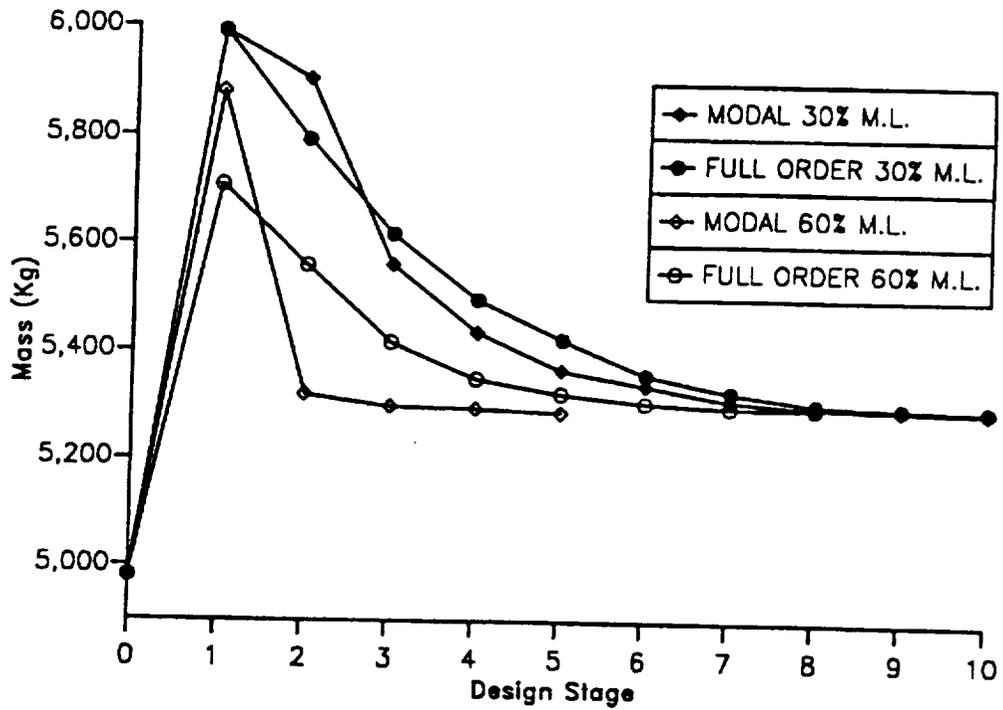
Figure 4. Mass Iteration Histories for Example 1
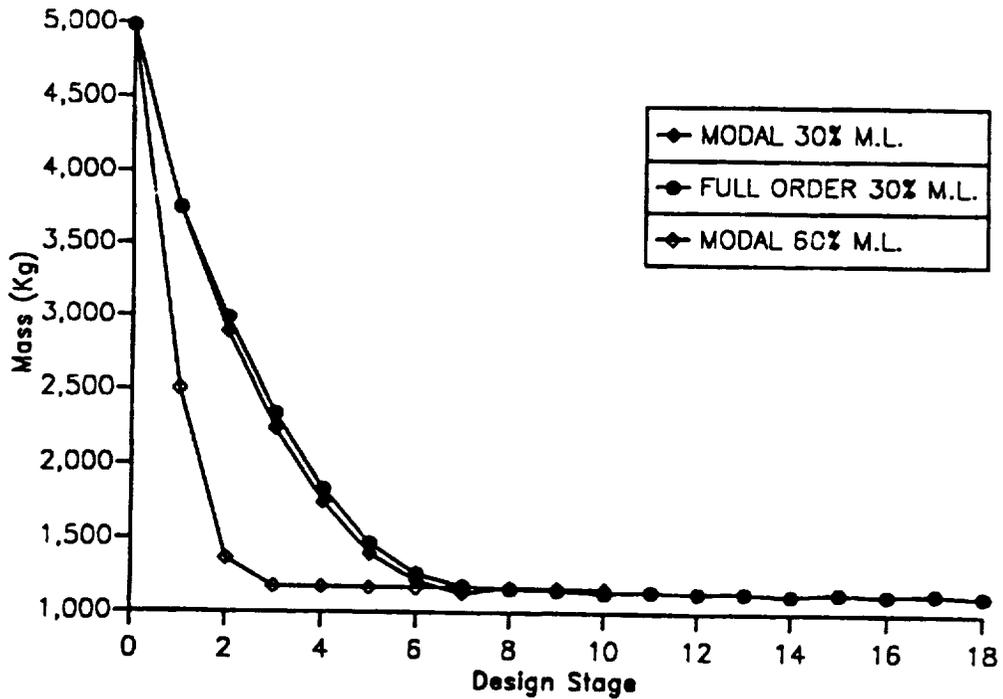
Figure 5. Mass Iteration Histories for Example 2

EFFICIENT EIGENSOLUTION REANALYSIS OF NONCLASSICALLY DAMPED STRUCTURES

Bo Ping Wang
Department of Mechanical Engineering
The University of Texas at Arlington
Box 19023
Arlington, Texas 76019

INTRODUCTION

Effective methods of approximate eigensolution reanalysis of modified nonclassically damped structures are developed in this paper. For structures with passive or active discrete damping devices or with damping treatment, the system becomes non-proportionally damped and the computation of its dynamic responses may require the use of complex modes. For larger systems, the computation of complex modes is very expensive. Thus it is desirable to have approximate reanalysis techniques for the efficient evaluation of the effect of design changes.

In recent years, the assumed mode reanalysis method has been successfully applied to minimum weight design of undamped structures with natural frequency constraints [1]. The accuracy of the assumed mode reanalysis method can be improved dramatically if the global approximation function includes the normal modes of the original system and their derivatives [2]. This approach has been demonstrated to be effective even for a system with shape changes. In this paper, the approach used by Noor et al. [2] for eigensolution reanalysis of undamped structures will be extended to treat a nonclassically damped system.

EIGENVALUE PROBLEM FOR DAMPED STRUCTURES

Consider a linear structure with general viscous damping. The equations for free vibration of the discrete system are

$$M\ddot{q} + C\dot{q} + Kq = 0 \tag{1}$$

where M, C, and K are mass, damping and stiffness matrices respectively, and q is the displacement vector. Assuming these matrices are symmetric, the system of second order ordinary differential equations can be expressed as:

$$B_1\dot{x} - A_1x = 0 \tag{2}$$

where

$$x = \begin{Bmatrix} q \\ \dot{q} \end{Bmatrix} \tag{3}$$

and

$$B_1 = \begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix} \tag{4}$$

$$A_1 = \begin{bmatrix} 0 & -K \\ -K & -C \end{bmatrix} \tag{5}$$

The eigenvalue problem corresponding to Eq. (2) is

$$\lambda_i B_1 u_i = A_1 u_i \tag{6}$$

or

$$A u_i = \lambda_i u_i \tag{7}$$

where

$$A = B_1^{-1} A_1 \tag{8}$$

provided B_1^{-1} exists.

Assuming all the eigenvalues of Eq. (6) or (7) are distinct, then, it can be shown that the eigenvalues satisfy the following orthogonality conditions:

$$u_j^T A_1 u_i = u_j^T B_1 u_i = 0 \tag{9}$$

for $i \neq j$.

It should be noted that Eq. (9) is true since A_1 and B_1 are symmetric matrices.

When the system is being modified, its mass, stiffness and damping matrices are changed by ΔM, ΔK and ΔC respectively. The new eigenvalue problem is given by

$$\lambda_i' B_1' u_i' = A_1' u_i' \tag{10}$$

where

$$A_1' = A_1 + \Delta A_1 \tag{11}$$

$$B_1' = B_1 + \Delta B_1 \tag{12}$$

and

$$\Delta A_1 = \begin{bmatrix} 0 & -\Delta K \\ -\Delta K & -\Delta C \end{bmatrix} \tag{13}$$

$$\Delta B_1 = \begin{bmatrix} -\Delta K & 0 \\ 0 & \Delta M \end{bmatrix} \tag{14}$$

For an n-degree of freedom system, the eigenvalue problems of Eqs. (6) and (10) are of order 2n. For underdamped structures, the eigensolution includes n pairs of complex conjugate eigenvalues with n pairs of eigenvectors that are complex conjugate to each other. Assuming L pairs of eigensolutions are computed for the nominal structure (i.e. Eq.

(6)), the reanalysis problem is to use this information to solve the eigenproblem of the modified system (Eq. (10)).

EFFICIENT EIGENSOLUTION REANALYSIS METHODS

For undamped structures, efficient reanalysis has been studied quite extensively by many researchers [1-4]. For local modification, a receptance-based reanalysis method for a general damped system has been developed [4]. In this paper, several reanalysis approaches based on the classical Bubnov-Galerkin approach will be presented.

Substituting Eqs. (11) and (12) into (10) the following equation is obtained after we split the operator:

$$(\lambda_i' B_1 - A_1) u_i' = -\lambda_i' \Delta B_1 u_i' + \Delta A u_i' \tag{15}$$

Let

$$\lambda_i' = \lambda_i + \Delta\lambda_i \tag{16}$$

Equation (15) can be written as

$$\lambda_i' (B_1 - A_1) u_i' = f_i \tag{17}$$

where

$$f_i = -\lambda_i' \Delta B_1 u_i' + \Delta A_1 u_i' - \Delta\lambda_i B_1 u_i' \tag{18}$$

It should be noted that Eq. (17) is the exact eigenvalue problem of the modified system written in a different form. For approximate solution, let us substitute $\Delta\lambda_i = \Delta\widetilde{\lambda}_i$, $\lambda_i' = \widetilde{\lambda}_i = \lambda_i + \Delta\widetilde{\lambda}_i$, $u_i' = u_i$ into f_i to yield

$$(\lambda_i B_1 - A_1) u_i' = \widetilde{f}_i \tag{19}$$

where

$$\widetilde{f}_i = -\widetilde{\lambda}_i \Delta B_1 u_i + \Delta A_1 u_i - \Delta\widetilde{\lambda}_i B_1 u_i \tag{20}$$

It should be noted that $\Delta\widetilde{\lambda}_i$ can be estimated from

$$\widetilde{\Delta\lambda}_i = u_i^T (\Delta A_1 - \lambda_i \Delta B_1) u_i \tag{21}$$

and the eigenvectors are normalized so that

$$u_i^T B_1 u_i = 1 \tag{22}$$

The general solution of Eq. (19) is then

107

$$u_i' = u_i + \Delta u_i \tag{23}$$

where Δu_i is the particular solution of the system of equations (19). It should be noted Δu_i can be interpreted as a derivative of the eigenvector u_{ij} and can be solved by several techniques, such as Nelson's method [5].

Equation (23) is the explicit approximate reanalysis for the ith eigenvector. Once u_i' is available, the corresponding eigenvalue can be computed from

$$x_i' = \frac{{u_i'}^T A_1' u_i'}{{u_i'}^T B_1' u_i'} \tag{24}$$

Another approach of solving the reanalysis problem is to assume

$$u_i' = T \, \eta_i \tag{25}$$

The modified eigenvalue problem then becomes

$$\lambda_i' \, B_1^* \, \eta_i = A_1^* \, \eta_i \tag{26}$$

where

$$B_1^* = T^T \, B_1' \, T \tag{27}$$

$$A_1^* = T^T \, A_1' \, T \tag{28}$$

The accuracy of the above approach depends on the choice of basis vectors in the transformation matrix T. The simplest choice of T would be a set of truncated eigenvectors, that is

$$T = [u_1 \, u_2 \cdots u_L] \tag{29}$$

In view of Eq. (23), one may incorporate Δu_i and form T as

$$T = [u_1 \cdots u_m, \, \Delta u_1 \cdots \Delta u_L] \tag{30}$$

Since the computation of Δu_i by Nelson's method is quite involved, the following scheme may be used to find an approximate $\overline{u}_i$ for Δu_i. Let us add $\alpha \lambda_i B_1 u_i'$ to both sides of Eq. (17) and then evaluate the right side as before to get the following equations for $\overline{\Delta u_i}$

$$((1 + \alpha)\lambda_i \, B_1 - A_1) \overline{\Delta u_i} = \tilde{f}_i - \alpha \, \lambda_i \, B_1 \, u_i \tag{31}$$

where α is a small positive number. In the numerical example, a value of 0.01 is used for α. Note that $\overline{\Delta u_i}$ is an approximation to the eigenvector derivative. For numerical stability, the contribution of lower modes to $\Delta \overline{u}_i$ should be filtered out. The resulting

vector $\overline{\Delta u_i}$ can be considered as a "residual mode" of the system. The transformation matrix T can then be constructed as

$$T = \begin{bmatrix} u_1...u_L, & \widetilde{\Delta u_1}...\widetilde{\Delta u_L} \end{bmatrix} \tag{32}$$

In summary, the following eigensolution reanalysis methods are available.

I.   Explicit Method

Use Eq. (23) to compute approximate u_i' and then use Eq. (24) to compute λ_i'.

II. Implicit Methods

The implicit methods involve the solution of Eq. (26) for λ_i' and η_i and then use Eq. (25) to compute u_i'. Three variations are available.

   (1)   Assumed mode: Use Eq. (29) for the transformation matrix. This method is designated as Ai in Table 3, where i is the number of pairs of modes used.

   (2)   Improved assumed method using Eigenvector Derivatives: Use Eq. (30) for the transformation matrix. This method is designated as AIi in Table 3, where i is the number of pairs of modes used.

   (3)   Improved assumed mode method using Approximate Eigenvector Derivatives: Use Eq. (32) as the transformation matrix. This method is designated as AAi in Table 3, where i is the number of pairs of modes used.

The above methods are applied to a numerical example in the next section.

NUMERICAL EXAMPLE

The ten degree of freedom mass-spring-damper system shown in Figure 1 is used to test the reanalysis methods proposed in this paper. Five cases are studied. They are defined in Table 2. These cases range from uniform change in stiffness and damping properties (Case A) to more severe modifications that include the removal of one of the support springs (Case D) and the removal of all dampers (Cases E).

The first two eigenvalues of the modified systems are summarized in Table 3. These include the exact solution and reanalysis results by several methods.

For Case A and C, all reanalysis methods perform well. For cases B and E, the assumed mode method requires the use of two pairs of modes to yield reasonable estimates of the fundamental eigenvalue. For case D, which involves the removal of spring No. 11 in the model, the assumed mode method performs poorly, even with four pairs of modes. On the other hand, the improved methods yield good results for all test cases when using one pair of modes plus the associated residual modes.

CONCLUDING REMARKS

Several methods for approximate reanalysis of damped systems are presented in this paper. For systems with moderate modification, all methods yield reasonable results. For systems with extensive modification, the reanalysis results may indicate a bogus unstable eigenvalue (i.e. eigenvalue with positive real part). Thus, the reanalysis method should be used with caution. The improved assumed mode methods provide significant improvement over the assumed mode method. These reanalysis formulations can be used to construct high quality approximate models for optimum design of damped structures.
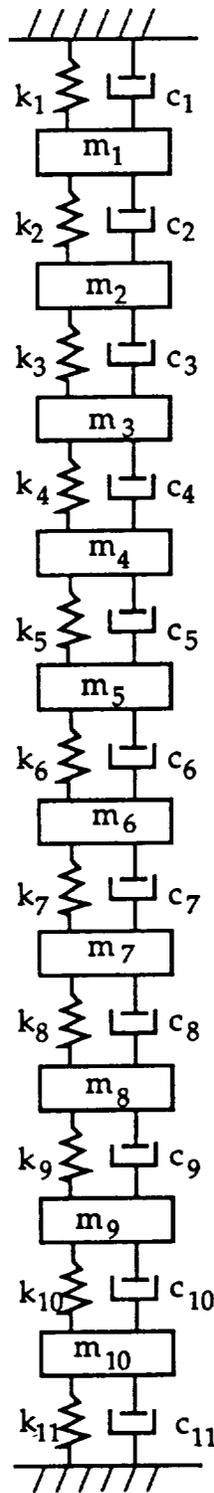
Figure 1. Ten Degree of Freedom Mass-Spring-Damper System

Table 1. Data for Numerical Example

i	m_i	k_i	c_i
1	1	1000	5
2	2	2000	10
3	3	3000	10
4	4	4000	15
5	5	5000	15
6	2	6000	10
7	3	7000	5
8	4	10000	20
9	2	30000	25
10	5	20000	25
11	-	30000	0

Table 2. Definition of Cases

Case	Modifications		
	Mass	Stiffness	Damping
A	None	$\Delta k_i = 0.25\, k_i$	$\Delta c_i = -0.3\, c_i$
B	None	$\Delta k_1 = 5000$ $\Delta k_6 = 10000$	$\Delta c_2 = 10$
C	$\Delta m_3 = 52$ $\Delta m_9 = 208$	None	None
D	None	$\Delta k_{11} = -30000$	$\Delta c_i = -0.3\, c_i$
E	None	$\Delta k_1 = 5000$ $\Delta k_6 = 10000$	$\Delta c_i = -c_i$

112

Table 3. Summary of Results for Ten Degree of Freedom Example

Method	Case A		Case B		Case C		Case D		Case E	
	Eigenvalues									
	λ_1	λ_3	λ_1	λ_3	λ_1	λ_3	λ_1	λ_3	λ_1	λ_3
Exact	-0.1120 +12.1852 i	-0.8109 + 29.8325 i	-0.2329 +13.3156 i	-1.2569 +29.6443 i	-0.0319 +4.6353 i	-0.0382 +8.2588 i	-0.0304 +4.5219 i	-0.2845 +19.2199 i	0 + 13.32 i	0 + 29.64 i
A1	-0.1118 +12.1852 i	N/A	-0.1301 +16.1283 i	N/A	-0.0453 +4.8547 i	N/A	-0.0853 +10.1106 i	N/A	0.0343 +16.1286 i	N/A
AI1	-0.1120 +12.1860 i	N/A	-0.2298 +13.1868 i	N/A	-0.0359 +4.6323 i	N/A	-0.2170 +4.8847 i	N/A	-0.0094 +13.1863 i	N/A
AA1	-0.1119 +12.1852 i	N/A	-0.2306 +13.1906 i	N/A	-0.0367 +4.6329 i	N/A	-0.1429 +5.0390 i	N/A	-0.0108 +13.1901 i	N/A
A2	-0.1097 +12.186 i	-0.8373 +29.8274 i	-0.3388 +14.4759 i	-0.8601 +35.8799 i	-0.0211 +4.7237 i	0.3967 +12.0897 i	-0.1087 +10.1478 i	-0.6795 i +25.6470 i	-0.1613 +14.4693 i	0.3011 +35.8830 i
AI2	-0.1119 +12.1852 i	-0.8113 +29.8326 i	-0.02328 +13.3294 i	-1.2638 +29.6410 i	-0.0327 + 4.6376 i	-0.0613 +8.4144 i	-0.2676 +4.3441 i	-0.2988 +19.1666 i	0.0005 +13.3303 i	-0.0278 +29.6216 i
AA2	-0.1120 +12.1852 i	-0.8110 + 29.8330 i	-0.2329 +13.3343 i	-1.2633 +29.6476 i	-0.0316 +4.6383 i	-0.0898 + 8.4778 i	0.2884 +4.5939 i	-0.2930 +19.2004 i	-0.0008 +13.3354 i	-0.0349 +29.8281 i
A4	-0.1093 +12.1806 i	-0.8093 + 29.8357 i	-0.4784 +14.0547 i	-1.9704 +32.1478 i	0.0166 +4.7997 i	0.2223 +10.0760 i	0.5525 +10.5834 i	-0.2326 +26.0879 i	-0.2682 +14.0314 i	-1.0771 +32.0315 i

113

REFERENCES

1. Wang, B. P., "Structural Dynamic Optimization Using Reanalysis Techniques," <u>The International Journal of Analytical and Experimental Modal Analysis</u>, Vol. 2, No. 1, pp. 50-58, January, 1987.

2. Noor, A. K. and Whitworth, S. L., "Reanalysis Procedures for Large Structural Systems," <u>International Journal for Numerical Methods in Engineering</u>, Vol. 26, 1929-1748 (1988).

3. Wang, B. P., Caldwell, S. and Smith, C.M., "Improved Eigensolution Reanalysis Procedures in Structural Dynamics," Paper No. 46, <u>Proceeding of 1990 MSC World Users Conference</u>.

4. Wang, B. P., Pilkey, W. D., and Palazzolo, A. B., "Reanalysis, Modal Synthesis and Dynamic Design", Chapter 8 of <u>State-of-the-Art Surveys on Finite Element Technology</u>, edited by A. Noor and W. Pilkey, ASME, New York 1983.

5. Nelson, R. B., "Simplified Calculation of Eigenvector Derivatives", <u>AIAA J.</u>, 14, 1201-1205 (1976).

NUMERICAL PROPULSION SYSTEM SIMULATION

John K. Lytle
David A. Remaklus
Lester D. Nichols

*NASA Lewis Research Center
Cleveland, OH*

I. Abstract

The cost of implementing new technology in aerospace propulsion systems is becoming prohibitively expensive. One of the major contributors to the high cost is the need to perform many large scale system tests. Extensive testing is used to capture the complex interactions among the multiple disciplines and the multiple components inherent in complex systems. The objective of the Numerical Propulsion System Simulation (NPSS) is to provide insight into these complex interactions through computational simulations. This will allow for comprehensive evaluation of new concepts early in the design phase before a commitment to hardware is made. It will also allow for rapid assessment of field-related problems, particularly in cases where operational problems were encountered during conditions that would be difficult to simulate experimentally. The tremendous progress taking place in computational engineering and the rapid increase in computing power expected through parallel processing make this concept feasible within the near future. However it is critical that the framework for such simulations be put in place now to serve as a focal point for the continued developments in computational engineering and computing hardware and software. The NPSS concept which is described below will provide that framework.

II. Introduction

The traditional design and analysis procedures applied to complex systems decomposes the system into isolated disciplines and components to reduce their complexity. Consequently, the interactions between disciplines and components is limited by the amount of interaction between individuals or teams working the problem. When severe demands are placed on the size, weight, and performance of the system, then the designs by nature become highly integrated with tight coupling between disciplines and components. The tight coupling can result in unforseen interactions which would produce unsatisfactory system performance. If the coupling is not resolved until the system has been built and tested, then the system must undergo redesign and retesting. Typically several iterations of the design-build-test cycle are required before desired performance is achieved. This is an extremely costly and time consuming process. As a result, the introduction of advanced technology takes many years as these systems slowly evolve. Pressure exists to reduce the time and cost associated with introducing new technology. This can be achieved through optimizing existing design practices and through introducing a higher level of concurrent engineering into the design process.

NPSS is a top-down systems approach which would provide designers with a tool to incorporate the relevant factors which affect system performance early in the design and analysis process when changes or modifications can be made relatively inexpensively. In terms of a propulsion system, such as an air-breathing gas turbine engine, this means coupling of disciplines and components computationally to determine system attributes such as performance, reliability, stability and life. Since these system attributes have traditionally been obtained in the test cell, NPSS is referred to as the "numerical test cell". A complete system analysis which includes multiple disciplines is a computationally intensive task requiring a high performance computing platform including massively parallel processors and a user interface consisting of

115

expert systems, data base management systems and visualization tools. These essential elements of NPSS are depicted in Figure 1. The integrated, interdisciplinary system analysis requires advancements in the following technologies: 1. interdisciplinary analysis to couple the relevant disciplines such as aerodynamics, structures, heat transfer, chemistry, materials, controls; 2. integrated system analysis to couple subsystems, components and subcomponents at an appropriate level of detail; 3. a high performance computing platform composed of a variety of architectures, including massively parallel processors, to provide the required computing speed and memory; and 4. a simulation environment that provides a user-friendly interface between the analyst and the multitude of complex codes and computing systems that will be required to perform the simulations.

The implementation and integration of these technologies is a major challenge. The simulation environment and integration capabilities are depicted in Figure 2. The NPSS system simulation is represented by the horizontal bar to signify that NPSS integrate into a system simulation the advancements that will continue to take place in the single discipline, component and computing fields. In this way, NPSS will provide a focus for research and development in the disciplines, components and computing fields. An additional challenge in NPSS will be the formation of interdisciplinary teams across NASA, industry, universities and other government agencies to develop and implement the needed technologies.

This paper describes the approach being developed at the NASA Lewis Research Center to address the issues of high-fidelity propulsion system computational simulations. The focus is on system simulation, interdisciplinary analysis, simulation environments, and parallel computing.

III. Approach

A. System Simulation

The computational system simulations will be based on the view that only phenomenon that affects system attributes, such as life, reliability, performance and stability of a propulsion system, is of interest to the designer or analyst. In addition, detailed analyses of an entire propulsion system will be so complex that even computers of teraFLOPS speed will not be sufficient to perform cost effective computations. Consequently, a framework is being developed that will allow the physical processes resolved from a detailed analysis of a component or subcomponent to be communicated to a system analysis performed at a lower level of detail for purposes of evaluating system attributes. Conversely, the system analysis will provide the ability to evaluate which physical processes occurring on the component and subcomponent level are important to system performance. This will allow the engineer or scientist to focus or "zoom in" on the relevant processes within components or subcomponents. The zooming concept is depicted in Figure 3. In this particular example, a detailed analysis of the fan would be performed to study, for example, the effect of a new blade design on system performance. The inlet and compressor would be modeled a slightly lower levels of fidelity to resolve phenomenon such a inlet distortion or upstream influences of the compressor blading. The combustor, turbine and nozzle would be modeled at less detail, perhaps to determine shaft horsepower and thrust.

The implementation of the zooming approach requires a hierarchy of codes and models to be in place to provide a wide range of capabilities from detailed three-dimensional, transient analysis of components to time- and space- filtered analysis of the subsystems and systems. Modeling approaches will be developed for communicating information from a detailed analysis to a filtered analysis. This will require additional research in understanding the mechanisms by which phenomenon on different length and time scales communicate. Research already underway in computational fluid

116

dynamics and structural mechanics to develop this modeling approach will be extended to consider processes and scales appropriate for the entire propulsion system.

The fluid dynamic simulation model that will serve as the basis for the integrated system model will be the Adamczyk [1] average-passage formulation which has been developed for multistage turbomachinery analysis. The average-passage is based on the filtered forms of the Naiver-Stokes and energy equations. This model was designed to resolve only the temporal and spatial scales that have a direct impact on the relevant physical processes. The effects of the unresolved scales, which appear as body forces and energy sources in the equations, are determined through semi-empirical relations which are based on results from physical experiments or high-resolution numerical simulations. The results from the lower resolution analysis appear as boundary conditions for the high-resolution simulations. This model is currently applicable to time- and space- averaging of phenomenon on the scale of the blade passing frequency and passage size. Further development is required to extend the model for filtering in the presence of multiple scales and for other system components.

The structures modeling will be aimed at developing a comparable computational capability that will provide a means to traverse multiple scales of spatial resolution with a minimum number of variables at each level. In this way, an analysis can proceed from a blade to a rotor to an engine core to the complete engine. The resulting system will have a minimum number of degrees of freedom consistent within the objectives of the analysis and will minimize the computational requirements. The methodology will be applicable to the solution of linear and incremental nonlinear analysis problems. This capability will be achieved through the formulation and implementation of a progressive substructuring technique [2].

B. Interdisciplinary Analysis

Aerospace propulsion systems are complex assemblies of dynamically interacting disciplines. The traditional approach is to handle the interactions by single disciplines in a sequential manner where one discipline uses information from the preceding calculation of another discipline. This is a lengthy, tedious, and often times, inaccurate approach. The alternative to this approach is using multidisciplinary coupling on a more fundamental level. A hierarchical approach will be employed that will reduce the dimensionality of the system while still retaining the essential system behavior. A variety of techniques will be evaluated for coupling discipline variables for selected propulsion system subcomponents, components and subsystems. These include the traditional sequential iteration, specially-derived matrices, and coupling at the fundamental equation level. All three methods will be applied to the filtered Naiver-Stokes and progressively substructured formulations with space and time scales that are consistent with the physics of the phenomenon being simulated. This approach differs from the classical analytical approach which minimizes the number of variables retained in the governing equations by using formal applied mathematical techniques. The proposed approach retains all of the primitive variables in the primitive equations. Sensitivity relations will be used to scope the degree of coupling and to decide on a solution strategy. Requisite technology base required for the development and definition of sensitivity relations includes: advanced methods of matrix operations for integration, differentiation, inversion and eigen value extraction, adaptive matrix partitioning, transfer matrices, and symbolic operators.

An essential element of complex system analysis is optimization. Optimization of complex systems involving multiple disciplines and components require streamlined algorithms for rapid solutions. Five different types of optimization algorithms will be developed: 1) hierarchical, 2) multi-scale, 3) multi-region, 4) multi-objective, and 5) adaptive. The hierarchical algorithms will provide the capability to select

dominant variables/disciplines/components during the optimization process. These variables/disciplines/components will continually change as the optimization progresses. The multiscale algorithms will provide the formalism for opitimizing at different scales as the optimization progresses. These algorithms will allow local optimization simultaneously with global but at different rates and with different accuracy. The multi-region algorithm will be similar to that for multi-scale but structured for regions and components. That is, different regions/components can be optimized at different rates while the rates can change as the system optimum becomes more sensitive to critical regions/components. The multi-objective algorithm will handle the simultaneous optimization of multidisciplinary, multicomponent problems. Formalisms will be included for coupled objectives and/or weighted objectives as well as discriminatory selection for a critical discipline/component. The adaptive algorithms will have the logic to progressively monitor dominant conditions and to provide the hierarchical algorithm with information for selecting the appropriate variables/disciplines/components during the optimization process. The technology base to support development of these optimization algorithms include mathematical optimization techniques: linear, nonlinear, continuous, discrete, constrained, unconstrained, substructuring, variable linking as well as a variety of direct nonlinear mathematical and optimality criteria search methods that have evolved over the years [2].

C. Simulation Environment

The capability for users to simulate propulsion systems which include complex analyses on high performance, massively parallel computers will require extensive development of a user interface with a parallel/distributed computer implementation. The user interface will shield the user from the details of the system while providing sufficient guidance and assistance to perform the simulation at hand. The vision is that of totally "seamless" environment. The environment consists of the integration of physical sciences, computer sciences, computer systems software and computer systems hardware under the control of a global simulation executive. The computational simulation of multidiscipline, multicomponent problems consists of a large number of variables that require simultaneous solutions of multiscale, multiregion problems in local/global database environments. These types of problems can only be effectively solved in (massively) parallel processor computers and networks where distributed parallel programming concepts can be readily implemented. Logic and software will be developed to adaptively allocate solution strategies and processors for a single discipline and for interdisciplinary analysis of both the local and global levels. Construction of simulations is aided by a visual simulation editor coupled to an expert system "trained" in the use of the simulation codes. Artificial intelligence approaches, including expert systems and neural nets, will be investigated for assisting the user in making appropriate decisions in constructing a simulation. Advanced computer graphics, visualization and animation complete this environment.

An important feature of the simulation environment is that it provides modularity and flexibility. Modularity is important to facilitate the coupling of disciplines and components at various levels of detail and to facilitate insertion of new or proprietary codes. Flexibility is essential to efficiently expand the simulator with increasing software and hardware capabilities. This will be accomplished through the application of object oriented programming techniques. Consequently, a series of objects appropriate for and completely describing propulsion systems will be defined as will the inter-relationships between these objects. From this, a series of standardized interfaces will be developed consisting of both object data extraction and manipulation routines.

D. Parallel Computing

It is expected that advances in parallel computing will make the integrated, interdisciplinary analysis of complex systems practical in design and analysis environments. At the same time, it is expected that approaches to problem formulation and algorithm design will have to change to be able to exploit the new parallel architectures. Therefore, NPSS will establish a testbed environment so application and computer scientists can work closely together with state-of-the-art hardware and software tools to develop algorithms and to identify the appropriate computing architectures for the propulsion system applications.

The long-range goal of NPSS is to implement the shared memory model, in either hardware or software, on massively parallel platforms. In the shared memory model, the programmer sees a uniform programming platform even though the hardware platform may consist of a variety of architectures such as cubes, rings, etc. This not only simplifies the requirements for developing new code, but provides the easiest, most flexible platform for the conversion of serial FORTRAN code which proliferates the computational engineering community today. Much of this code is now and will continue to be useful in NPSS applications. While the shared memory model is a long range objective, other technologies will need to be evaluated in the near term.

The near-term goal for the testbed development is to acquire a relatively small parallel processor system of approximately 5 gigaFLOPS peak performance and 20 gigabytes of disk storage. The initial testbed, planned for 1992, will provide a dedicated platform to begin the parallelization of single discipline codes; develop coupling algorithms and interfaces, and to develop the zooming concept. These activities will culminate in the demonstration of a simulation of the High Speed Civil Transport propulsion system. The testbed is expected to be upgraded in 1994 to 15 gigaFLOPS peak performance with 30 gigabytes of disk storage. The upgraded testbed will be utilized to perform a high-fidelity demonstration of the High Speed Civil Transport with zooming capability on multiple components.

Parallel processing activities currently at the Lewis Research Center involve investigating architecture and algorithm compatibility issues on the Center's Hypercluster Testbed, porting of structural mechanics and fluid dynamics codes to shared memory machines such as the Alliant-FX/80 and the Cray YMP-8/464 and configuring transputer systems for use with structural dynamics codes. The Lewis Research Center is also supporting the development of a 2-dimensional grid parallel computing architecture at Mississippi State University. Activities are underway for the purchase of two Intel iPSC/860 8 processor machines in 1990. Additional hardware will be identified and procured in 1991 through the newly formed Advanced Computing Concepts Laboratory within the Computer Services Division at Lewis.

IV. Summary

The Numerical Propulsion System Simulation is a long range program with the ultimate goal of developing the capability of reducing the cost and time of developing advanced technology propulsion systems. This will be achieved through a cooperative effort of NASA, industry, universities and other Government agencies to develop the necessary technologies to integrate disciplines, components, and high performance computing into a user-friendly simulation environment. The technologies associated with the physical sciences must include model development that reflects an understanding of the relevant physical processes rather than brute-force computational analysis. The computational algorithms must be developed in concert with the computing architectures to ensure

efficient performance, particularly with highly and massively parallel processors. In addition, a strong and effective management team is required to form the interdisciplinary teams from all organizations that will be required to define, advocate, and implement these technologies.

V. Acknowledgements

VI. References

1. Adamczyk, J.J., "Model Equation for Simulating Flows in Multistage Turbomachinery," ASME 85-GT-226, 1985.

2. Chamis, C.C, Hopkins, D.A., and Berke, L., Personal Communications, NASA Lewis Research Center, Cleveland, OH, 1989.
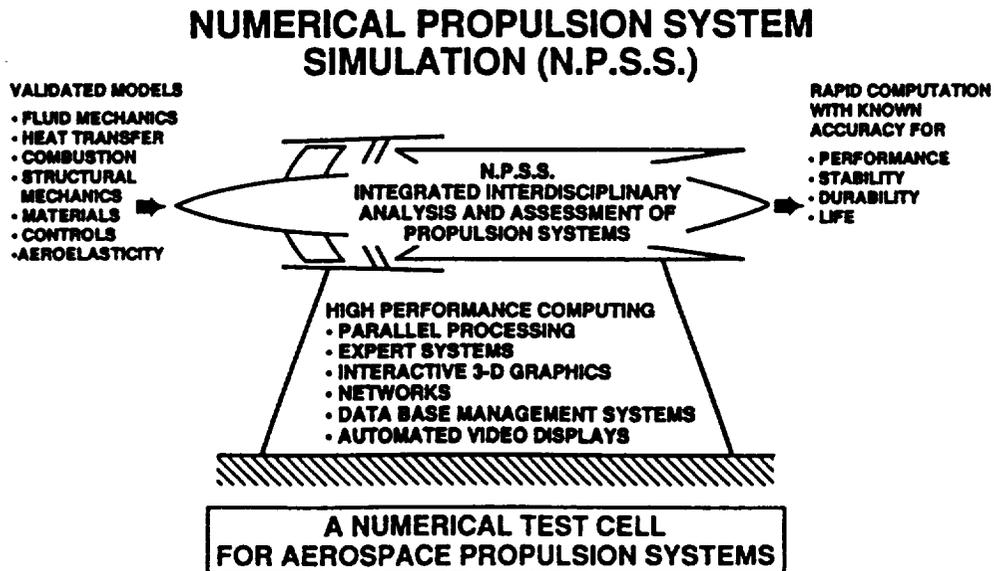
Figure 1. The Numerical Propulsion System Simulation is the concept of a "numerical test cell".

NUMERICAL PROPULSION SYSTEM SIMULATION INTEGRATION
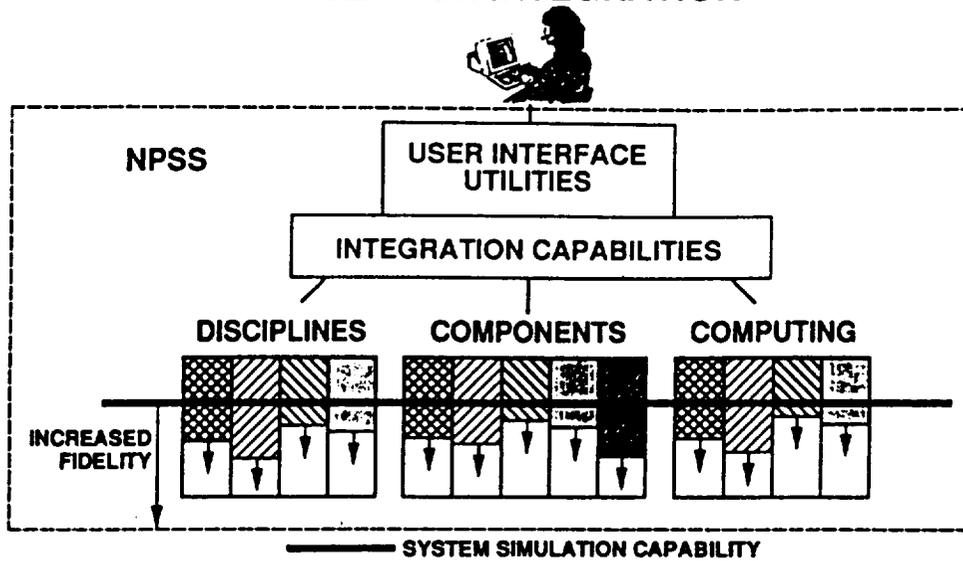
Figure 2. System simulation through NPSS will involve integration of disciplines, components and computing hardware and software.

NUMERICAL PROPULSION SYSTEM SIMULATION
"ZOOMING IN" ON FAN EFFECTS
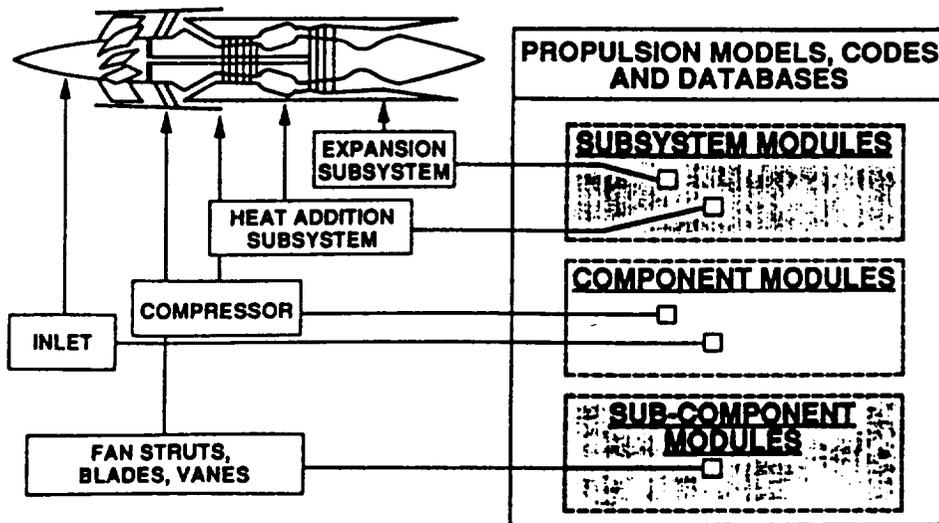
Figure 3. Cost-effective computations of complex propulsion systems will require the ability to vary the detail of analysis or "zoom".

INTEGRATED USE OF COMPUTER PROGRAMS FOR ANALYSIS AND OPTIMIZATION OF AIRCRAFT STRUCTURES

YOUNG IN MOON
WRDC/FIBRA/ASIAC, WRIGHT-PATTERSON AFB, DAYTON, OHIO 45433

1.0 ABSTRACT

The objective of this paper is to present results from investigating the integrated use of five computer programs: ANALYZE, ASTROS, NASTRAN, OPTSTAT and VAASEL for analysis and optimization of a given structure. The structure designated for study purposes was the F-15E vertical tail. The concept of integrated use is limited to the capability of using a NASTRAN structural model to run each of the other specified programs. This was actually accomplished in practice by converting a NASTRAN model of the designated structure into an appropriate analysis model. For optimization purposes, the torque-box of the F-15E vertical tail was used.

2.0 DESCRIPTION OF FINITE ELEMENT PROGRAMS

2.1 NASTRAN

Although NASTRAN (NASa STRuctural ANalysis) is not a multidisciplinary analysis program, it is the largest general-purpose finite element structural system in use today and provides effective solutions for a wide variety of applications. In this paper, NASTRAN was used to obtain nodal deflections and element stresses for specified static loads and to compute natural frequencies and corresponding mode shapes of the F-15E vertical tail.

2.2 ASTROS

ASTROS (Automated STRuctural Optimization System) is a finite element code written primarily for preliminary design of aerospace structures. The basic objective for developing ASTROS was "to provide a state-of-the-art design tool that integrates existing methodologies into a unified multidisciplinary package" (Ref.1). For example, it adapts NASTRAN input format for structural analysis. Many potential analysis capabilities are available in ASTROS. Those selected for this investigation include: (1) Static and modal analysis of the F-15E vertical tail using the NASTRAN finite element model, and (2) User-selected optimization of boron/epoxy skin elements for one surface of the tail model.

2.3 ANALYZE

ANALYZE was developed to analyze aerospace structures using only membrane elements (Ref.2). Thus, only translational degrees-of-freedom are allowed. Since rotational motions are not permitted, element representations of the structure are expected to be stiffer. Elements used in this code include axial-force bar elements, triangular and quadrilateral constant strain elements, and a shear-panel element to reduce the possibility of overestimating stiffness resulting from the use of only membrane elements for webs. The User's Manual (Ref.2) documents a comprehensive theoretical background for the program, and

provides necessary information for both preparation of input data and interpretation of results.

2.4 OPTSTAT

This computer code was written by the same authors who wrote ANALYZE (Ref.2). It was developed primarily for in-house research to optimize structural designs subjected to static loads. As OPSTAT uses the same types of element as ANALYZE, rotational degrees-of-freedom are not allowed. Reference 3 provides the theoretical background underlying the program. OPTSTAT optimizes structures by resizing every element in the structural model using allowable stresses as design constraints; thus, it does not permit user-selected optimization.

2.5 VAASEL

VAASEL (Vulnerability Analysisof Aerospace Structures Exposed to Lasers), as the name suggests, has been designed primarily for laser vulnerability analysis of structures and has multidisciplinary analysis capability (Ref. 4). This code was designed to maintain maximum compatibility with both NASTRAN and ASTROS, and utilizes the same framework as ASTROS. This means that a given problem input data deck should run on each program with identical output results expected.

3.0 MODEL FOR ANALYSIS

The NASTRAN structural model shown in Figure 1 consists of 39 BAR elements, 1,529 ROD elements, and 1,160 SHEAR elements, with 673 nodes.

The vertical tail is made of three materials; boron/epoxy for skins,aluminum for the honeycomb core,and titanium for the fore and aft spars and the root rib. A structural model was first constructed and converted into a NASTRAN model, and verified with the McAir model (Ref.5) by comparing its geometry,material properties, and deflections at points of interest under the given loading conditions. This NASTRAN model has served as a basis for developing models for other computer programs. After verification of the model, the torque-box section of the F-15E vertical tail was isolated and used for analysis and optimization instead of working on the full model. Integrated use of other computer programs was achieved by generating an input data deck by transformation of the NASTRAN bulk data. Fortran programs were written for this purpose.

3.1 LOADING CONDITIONS AND TYPES OF COMPARISONS

Static analysis was performed for the complete vertical tail using NASTRAN, ASTROS and ANALYZE for calculations of deflections and stresses. First, a 400-pound load was applied at the tip of the model, followed by application of the maximum panel load and the maximum inboard load (Ref.5). Next, the torquebox was isolated from the complete tail model and run under three loading conditions: a 400-pound tip load, the maximum panel and inboard loads.

For dynamic analysis, NASTRAN and ASTROS were used to obtain natural frequencies and mode shapes.

For optimization studies, ASTROS was used for user-selected elements optimization while overall optimization was achieved by OPTSTAT. The number of plies in each layer of the skin elements was computed for both user selected and overall optimizations and compared to the original structure.

4.0 RESULTS AND COMPARISONS

4.1 STATIC ANALYSIS COMPARISONS

When static models were run, NASTRAN and ASTROS predict identical deflections under all loading conditions mentioned for both the complete and torquebox models. Differences occurring in shear stresses computed by each program were noted, but were negligible. Comparison of results from ANALYZE with the other two programs show that ANALYZE predicts smaller, but very close, deflections than either NASTRAN or ASTROS. A typical deflection curve along 50% chord line is shown in Figure 2, where ANALYZE is shown to predict about 3 % less deflection at the tip. This small difference is caused by the type of element used in the ANALYZE program. The model is cantilevered with nodes points fixed along the root, and therefore elements around the root exhibit stress concentrations. Design engineers should pay close attentioin to these highly-stressed elements.

4.3 DYNAMIC ANALYSIS COMPARISONS

For dynamic analysis of the torque-box, both the inverse method of eigenvalue extraction (INV) and the tridiagonal reduction method (FEER) were applied in running the NASTRAN model. In ASTROS, the Given's method of tridiagonalization (GIV) was used. The ASTROS predictions of the first five frequences and corresponding mode shapes are almost identical to NASTRAN results, and differences are negligible.

4.4 OPTIMIZATION RESULTS

The finite element model for optimization has 137 elements for the skin, resulting in 548 global design variables, since there are four layers in each element. The skin thicknesses were selected as design variables when the torquebox was optimized for each of the two design loads (Ref. 5).
Optimization was carried out in two ways: user-selected elements optimization performed by ASTROS, and overall optimization using OPTSTAT. Representative results are presented in Figures 3 and 4, and Table 1. Element 39 shown in both figures was selected as a typical example to pictorially illustrate the optimization results for the torque-box skin. Figure 3 shows that the original element weight of 0.1352 lbs was reduced to 0.0728 lbs after optimization for the maximum panel load and to 0.0364 lbs for the maximum inboard load.
The original number of plies in 0, 90, +45, -45 degree directions is (10, 2, 7, 7), respectively. Differences are expected to exist between user-selected and complete structural optimizations for a given design loading condition.
For example in Figure 4, the number of plies in each direction for complete optimization is (12, 1, 1, 1) for the net panel load and (10, 1, 2, 2) for the inboard load. This set of results differs significantly from results given in Figure 3 for user-selected optimization, where the number of resulting plies are (5, 1, 4, 4) for the net panel load and (2, 1, 2, 2) for the inboard load.
A complete set of results for every element in the torque-box is given in Reference 6. Table 1 provides a sample of these results for elements 1 through 24.

124

5.0 CONCLUDING REMARKS

The F-15E vertical tail was either analyzed or optimized using five finite element computer programs. As ASTROS and VAASEL follow the same algorithm for static structural analysis, identical results for nodal deflections and element stresses were computed (Ref. 6). ASTROS and NASTRAN predicted almost identical results in both static and dynamic analysis. Static predictions from ANALYZE models were also very close to those from NASTRAN and ASTROS. From optimization studies of the torque-box model, differences exist between types of structural optimization method for a specified loading condition, and between design loading conditions. Therefore, from the results of optimization, one can observe the potential danger of arbitrarily selecting one or more of several design load conditions and, independently, a type of optimization to arrive at an "optimized" redesign configuration. Unless all critical design loading conditions are considered, such an assumption could result in a totally inadequate design.

REFERENCES

1. "Automated Structural Optimization System (ASTROS)," AFWAL-TR-88-3028, Vols. I, II, and III, Decembe 1988

2. Venkayya, V. and Tischler, V., "ANALYZE - Analysis of Aerospace Structures with Membrane Elements," AFWAL-TR-78-170, December 1978

3. Venkayya, V. and Tischler, V., "OPTSTAT - A Computer Program for the Optimal Design of Structures Subjected to Static Loads," TM-FBR-79-67, June 1979

4. "Vulnerability Analysis of Aerospace Structures Exposed to Lasers (VAASEL)," Vol. II - User's Manual, October 1988

5. "F-15 Design Loads," Report No. MDC A0833, Vol. V, McDonnell Douglas Corporation, March 1971

6. Moon, Y., "Analysis and Optimization of Aircraft Structures," ASIAC Report No. 989.1A, September 1989
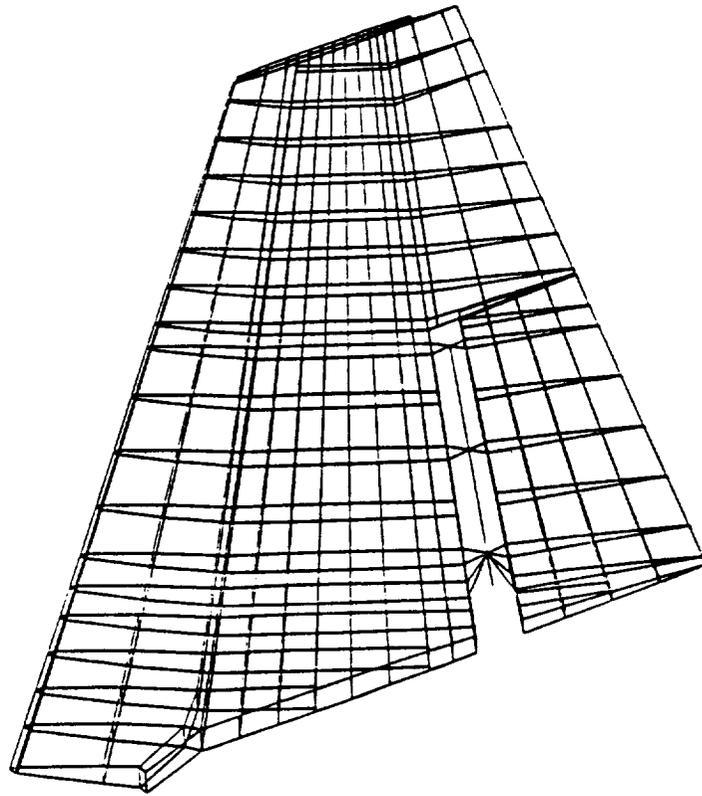
Figure 1 - Structural Model of F-15E Vertical Tail
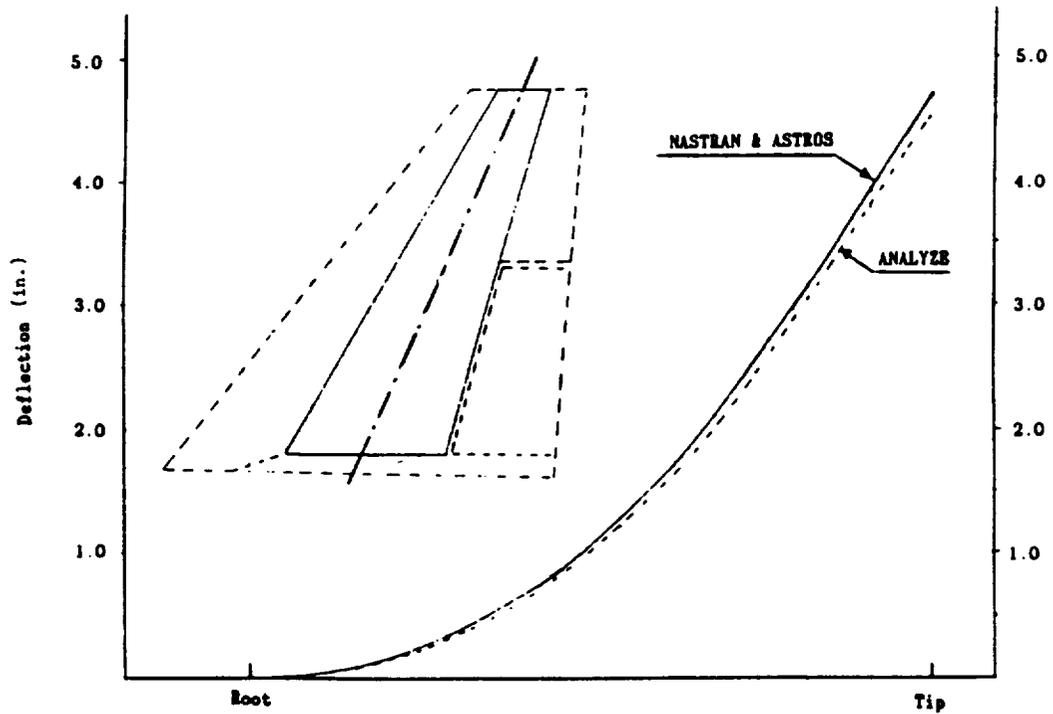
Figure 2 - Typical Deflection Curve Along 50% Chord Line

	number of plies				element weight
	0	90	+45	-45	
Original	10	2	7	7	0.1352
Panel Load	5	1	4	4	0.0728
Inboard Load	2	1	2	2	0.0364
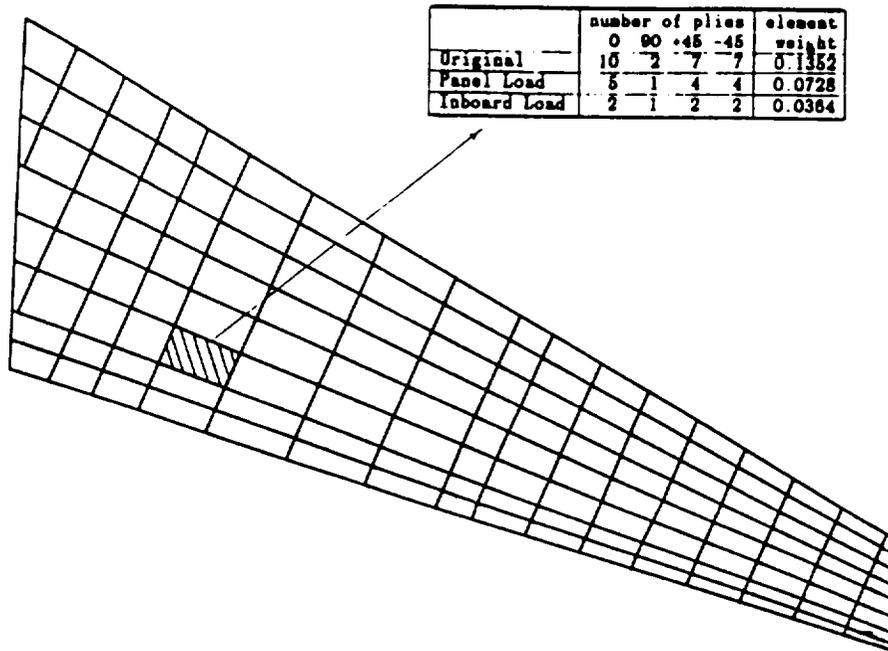
Figure 3 - User-selected Optimization : Outside Skin Elements Only

	number of plies				element weight
	0	90	+45	-45	
Original	10	2	7	7	0.1352
Panel Load	12	1	1	1	0.0780
Inboard Load	10	1	2	2	0.0780
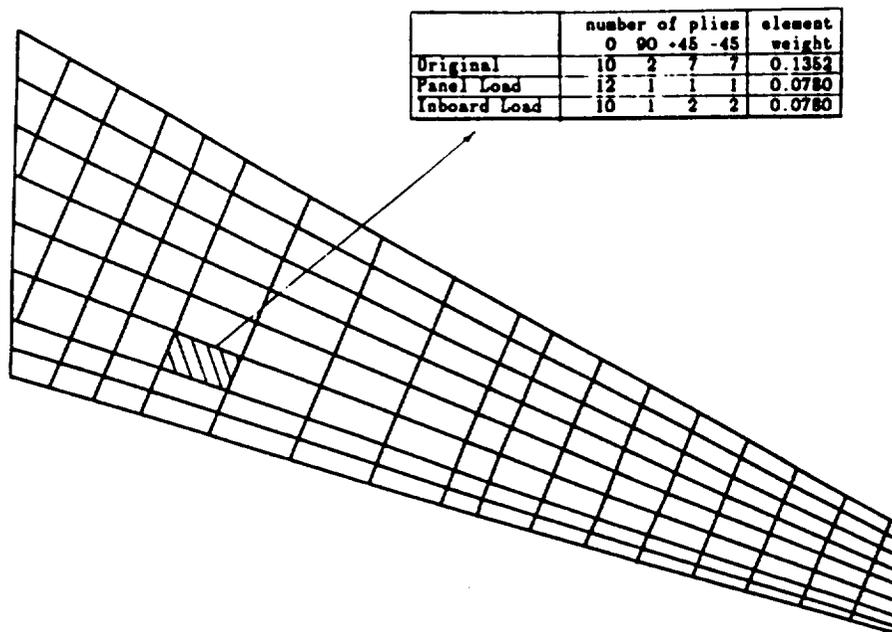
Figure 4 - Overall Optimization Skin Elements

Table 1 – Overall Optimization (OPTSTAT)

Elem. No.	Load	Ply counts in				Elem. No.	Load	Ply counts in			
		0	90	+45	-45			0	90	+45	-45
1	INITIAL	10	2	8	8	2	INITIAL	10	2	8	8
	PANEL	13	1	1	1		PANEL	12	1	1	1
	INBOARD	5	1	2	2		INBOARD	6	1	1	1
3	INITIAL	10	2	8	8	4	INITIAL	10	2	8	8
	PANEL	8	1	1	1		PANEL	7	1	1	1
	INBOARD	4	1	1	1		INBOARD	4	1	1	1
5	INITIAL	10	2	8	8	6	INITIAL	10	2	8	8
	PANEL	5	1	1	1		PANEL	5	1	1	1
	INBOARD	4	1	1	1		INBOARD	2	1	2	2
7	INITIAL	10	2	8	8	8	INITIAL	10	2	8	8
	PANEL	3	1	2	2		PANEL	3	1	2	2
	INBOARD	2	1	3	3		INBOARD	2	1	3	3
9	INITIAL	10	2	8	8	10	INITIAL	10	2	7	7
	PANEL	10	1	1	1		PANEL	9	1	1	1
	INBOARD	5	1	1	1		INBOARD	5	1	1	1
11	INITIAL	10	2	8	8	12	INITIAL	10	2	8	8
	PANEL	7	1	1	1		PANEL	7	1	1	1
	INBOARD	5	1	1	1		INBOARD	5	1	1	1
13	INITIAL	10	2	8	8	14	INITIAL	10	2	8	8
	PANEL	6	1	1	1		PANEL	5	1	1	1
	INBOARD	5	1	1	1		INBOARD	3	1	2	2
15	INITIAL	10	2	8	8	16	INITIAL	10	2	8	8
	PANEL	13	1	1	1		PANEL	10	1	1	1
	INBOARD	7	1	1	1		INBOARD	6	1	1	1
17	INITIAL	10	2	8	8	18	INITIAL	10	2	7	7
	PANEL	11	1	1	1		PANEL	9	1	1	1
	INBOARD	6	1	1	1		INBOARD	6	1	1	1
19	INITIAL	10	2	7	7	20	INITIAL	10	2	7	7
	PANEL	7	1	1	1		PANEL	7	1	1	1
	INBOARD	5	1	1	1		INBOARD	5	1	1	1
21	INITIAL	10	2	7	7	22	INITIAL	10	2	8	8
	PANEL	19	1	7	7		PANEL	11	1	6	6
	INBOARD	8	1	6	6		INBOARD	6	1	5	5
23	INITIAL	10	2	8	8	24	INITIAL	10	2	8	8
	PANEL	7	1	4	4		PANEL	12	1	6	6
	INBOARD	6	1	4	4		INBOARD	7	2	6	6

Structural Optimization With Constraints From Dynamics in LAGRANGE

F.Pfeiffer, G.Kneppe, C.Ross

Nomenclature

a	amplitude	q	vector of transformed coordinates	
$a_{i,max}$	maximum value for amplitude	x	vector of design variables	
C	matrix of expansion coefficients	z	element of x	
D	damping matrix	Y	transformation matrix	
F	tranformed matrix in frequ.resp.	y	complex displacemcent vector	
f	vector of external forces	y_{L_0}	Lanczos starting vector	
$\bar{f}$	assemblage of external forces	$\bar{y}$	eigenvector	
f	eigenfrequency	z	criterion vector for pursuing mode shapes	
$f_{i,max}$	upper bound for frequency	δ_{ij}	Kronecker symbol	
$f_{i,min}$	lower bound for frequency	ξ	modal damping coefficient	
g	constraint	λ	eigenvalue	
K	stiffness matrix	Ω	excitation frequency	
k_{ii}	normalization coefficient	Ω_l	lower bound for excitation frequency	
L	Cholesky decomposition of a matrix	Ω_u	upper bound for excitation frequency	
M	mass matrix	ω	angular frequency	
m	number of tranformation vectors			
m_{ii}	normalization coefficient	i, j, k, l	indices	
n	number of degrees of freedom	$(\ )^T$	transpose of vector or matrix	
nb	number of frequency bounds	$(\ )^k$	value in k-th iteration step	
p	number of feasible frequency intervals	i	$\sqrt{-1}$	

1 Introduction

Structural optimization problems are mostly solved under constraints from statics, such as stresses, strains or displacements under static loads. But in the design process dynamic quantities like eigenfrequencies or accelerations under dynamic loads become more and more important. Therefore, it is obvious that constraints from dynamics must be considered in structural optimization packages. This paper addresses the dynamics branch in MBB-LAGRANGE. It will concentrate on two topics, namely on the different formulations for eigenfrequency constraints and on frequency response constraints. For the latter the necessity of a system reduction is emphasized. The methods implemented in LAGRANGE are presented and examples are given.

2 Eigenfrequency Constraints

The main reason for formulating an eigenfrequency constraint in structural optimization problems is to avoid ranges in the frequency spectrum where excitation frequencies are known to occur under the working conditions of the structure.

To be able to formulate an eigenfrequency constraint the eigenfrequencies have to first be computed. Thus, the eigenvalue problem of the undamped structure

$$(K - \lambda M)\bar{y} = o \tag{1}$$

is considered. The eigenfrequencies f can be computed from the eigenvalues λ of (1) by

$$f = \frac{1}{2\pi}\sqrt{\lambda}. \tag{2}$$

Now eigenfrequency constraints can be formulated.

2.1 Upper and lower bounds for the i-th eigenfrequency

The most common case for an eigenfrequency constraint is given in figure 1, where a lower bound $f_{i,min}$ and an upper bound $f_{i,max}$ are given for the i-th eigenfrequency. These constraints can be written in a normalized form as
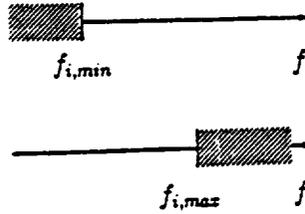
Figure 1: Common eigenfrequency constraint

$$g_{i,min} = 1 - \frac{f_{i,min}}{f_i} \geq 0 \tag{3}$$

$$g_{i,max} = 1 - \frac{f_i}{f_{i,max}} \geq 0. \tag{4}$$

The optimization algorithms which are most commonly used in structural optimization (e.g. SQP, SLP) require gradient information. By differentiating (3) or (4) with regard to a design variable $x \in x$ one has

$$\frac{\partial g_i}{\partial x} = \frac{\partial g_i}{\partial f_i} \frac{\partial f_i}{\partial \lambda_i} \frac{\partial \lambda_i}{\partial x}. \tag{5}$$

Thus, for computing the constraint derivative the eigenvalue derivative $\frac{\partial \lambda_i}{\partial x}$ must be computed which is given by

$$\frac{\partial \lambda_i}{\partial x} = \overline{y}_i^T \left(\frac{\partial K}{\partial x} - \lambda_i \frac{\partial M}{\partial x} \right) \overline{y}_i. \tag{6}$$

EXAMPLE 1 A propulsion stage of the Ariane launcher was discretized by a FE model with ca. 2500 degrees of freedom. A lower bound was given for the lowest eigenfrequency. The weight and constraint histories are shown in figure 2. The initial violation of the frequency constraint of ca. 3.8 % was completely eliminated. The weight was
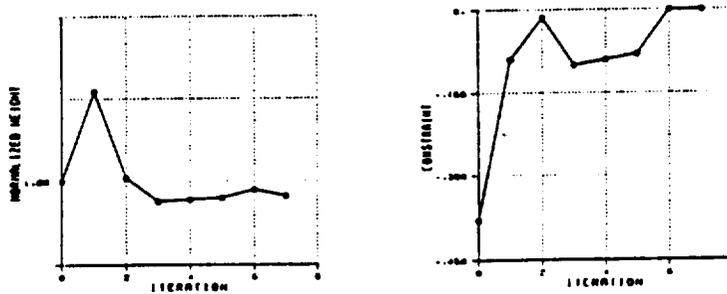
Figure 2: Weight and constraint history for Ariane propulsion stage

reduced by ca. 10 %. ∎

2.2 Formulation of a constraint for an eigenfrequency which corresponds to a given mode shape

In some applications, however, the formulations (3) and (4) for a fixed eigenfrequency index "*i*" are not sufficient. This shall be expounded by the following example.

EXAMPLE 2/I A satellite structure according to figure 3a) is regarded. In the initial analysis it was found that a characteristic axial mode shape exists for the inner structure (figure 3b)). An excitation of this initially 9th mode shape should be avoided, however, because the inner structure contains measurement equipment. Therefore, it was required that the eigenfrequency of this axial mode shape should be greater than a given lower bound. Some initial variations of the structure indicated that the corresponding mode shape changed between the 7th and the 10th eigenvector due to structural changes. Thus, putting a constraint on the 9th eigenvalue would have resulted in a completely wrong

130
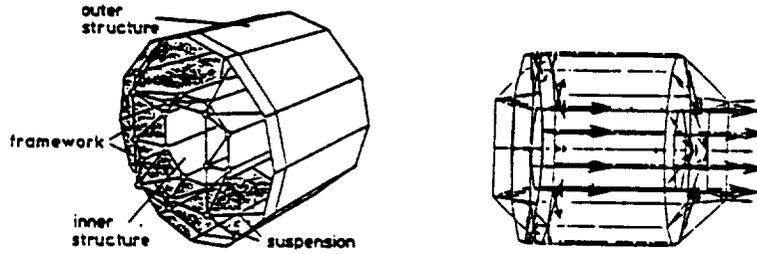
Figure 3: a)Satellite structure b) 9th mode shape (initial design)

design because the mode shape that was to be kept under control had changed its position in the eigenfrequency spectrum several times. ∎

Therefore, a strategy to pursue the given mode shape in consecutive iteration steps was developed. The strategy is based upon the M - or K - orthogonality of the eigenvectors

$$\overline{y}_i^T M \overline{y}_j = m_{ii}\delta_{ij} \qquad \overline{y}_i^T K \overline{y}_j = k_{ii}\delta_{ij}. \qquad (7)$$

To identify the i-th eigenvector of the k-th iteration step in the $(k+1)$-th step, the following vectors are defined:

$$z_i^k = L^{k^T} \overline{v}_i^k \qquad (8)$$

$$z_l^{k+1} = L^{k+1^T} \overline{y}_l^{k+1} \qquad (9)$$

where L is the Cholesky decomposition of either K or M. The required index "j" of the eigenvector in the $(k+1)$-th iteration step which corresponds to the i-th eigenvector of the k-th step is determined by

$$\frac{|z_i^{k^T} z_j^{k+1}|}{|z_i^k||z_j^{k+1}|} > \max_{\substack{l \\ l \neq j}} \frac{|z_i^{k^T} z_l^{k+1}|}{|z_i^k||z_l^{k+1}|} \qquad (10)$$

Graphically spoken, equation (10) states that the angle between the vector z_i^k and z_j^{k+1} has a minimum value (i.e. a maximum value for the scalar product in (10)) compared to the angles between z_i^k and any of the vectors z_l^{k+1} ($l \neq j$) of the $(k+1)$-th step.

EXAMPLE 2/II Using this algorithm the constraint for the satellite (figure 3a)) was formulated. Pursuing the given mode shape with the described strategy turned out to be very robust and gave correct results for other structures as well. The required frequency for the axial mode shape was obtained accurately while simultaneously minimizing the heat flux between the outer and the inner structure ([Kneppe89]). ∎

2.3 Multiple constraints for the i-th eigenfrequency

Another formulation, namely prescribing several bounds for the i-th eigenfrequency, may occur in some applications. For the case of two bounds the two possible variations are given in figure 4. Case a) (inclusion) can be handled with
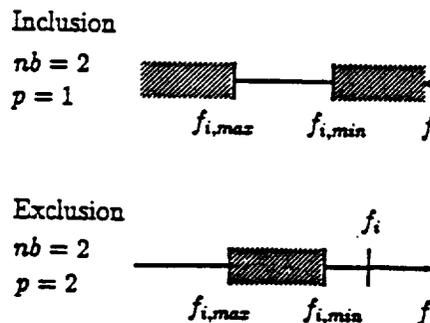
Figure 4: Two constraints for the i-th eigenfrequency

two constraints of type (3) and (4). Case b) (exclusion) cannot be described by the formulations given so far, however. If, for example. the eigenfrequency f_i lies in the right interval (figure 4b)), constraint (3) is satisfied whereas (4) is severely violated. Simultaneously satisfying both constraints is impossible in this case. Thus, a new formulation is required which is given by

$$g_i = (-1)^{p+1} \prod_{j=1}^{nb} g_{i,j} \geq 0, \tag{11}$$

where $g_{i,j}$ denote the constraints of the type (3) or (4), p the number of feasible frequency intervals and nb the number of bounds, respectively. For the exclusion case in figure 4b) the constraint can be written explicitly as ($p = 2, nb = 2$)

$$g_i = (-1) \left(1 - \frac{f_i}{f_{i,max}}\right) \left(1 - \frac{f_{i,min}}{f_i}\right) \geq 0, \tag{12}$$

which gives constraint values greater than zero for the two feasible frequency intervals.
Differentiation of (11) with regard to a design variable x yields

$$\frac{\partial g_i}{\partial x} = (-1)^{p+1} \sum_{j=1}^{nb} \frac{\partial g_{i,j}}{\partial x} \prod_{\substack{k=1 \\ k \neq j}}^{nb} g_{i,k} \tag{13}$$

which is a differentiable function in x (just as the constraint (11) itself).·
EXAMPLE 3 The lowest eigenfrequency of a clamped-free beam shall be constrained. The constraint is given by three bounds and can be sketched as in figure 5a). The initial design (ID) lies in an infeasible frequency interval. The
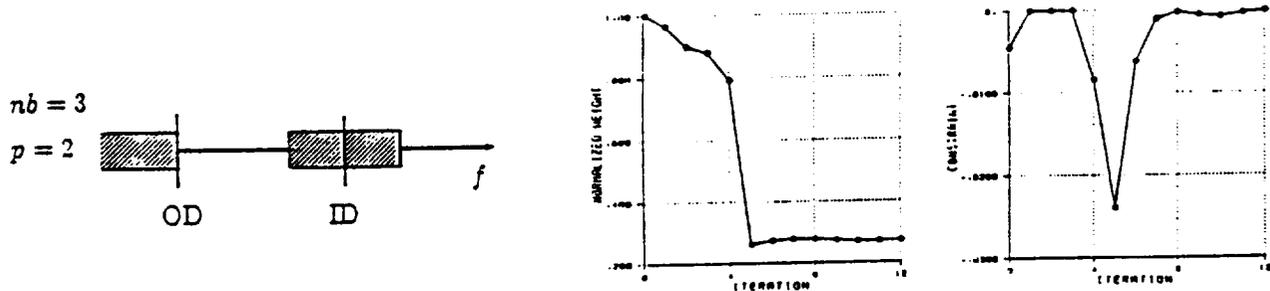
Figure 5: a) Multiple constraint for lowest eigenfrequency b) Weight history c) Constraint history

optimization gives an optimal design (OD) at the lower interval bound which is physically reasonable. Weight and constraint histories are shown in figure 5 as well. ■

3 Frequency Response Constraints

A more advanced type of constraints from dynamics is given by frequency response constraints. These may occur when a structure is harmonically loaded and it is required that the displacement, velocity or acceleration at certain points of the structure must not exceed prescribed values.
For simplicity, only displacement amplitudes a_i are considered in the following sections. Velocities and accelerations can be treated analogously.
In this case the constraint for the displacement amplitude in the i—th degree of freedom is given by

$$g_i = 1 - \frac{a_i(\Omega, x)}{a_{i,max}} \qquad \Omega_l \leq \Omega \leq \Omega_u, \tag{14}$$

where the excitation frequency Ω varies in a frequency interval given by its lower bound Ω_l and its upper bound Ω_u. The displacement vector y of a harmonically loaded structure (excitation frequency Ω) is computed by

$$(-\Omega^2 M + i\Omega D + K) y = f(\Omega). \tag{15}$$

$f(\Omega)$ denotes the vector of external forces. The amplitude a_i is computed from the complex displacement in the i—th coordinate as

$$a_i = \sqrt{\text{Re}(y_i)^2 + \text{Im}(y_i)^2}. \tag{16}$$

Equation (15) is a system of complex linear equations which is quite expensive to solve. To make a solution far less time-consuming one generally introduces a transformation

$$y = Yq. \tag{17}$$

The transformation matrix Y is a $n \times m$−Matrix ($m \ll n$) which reduces the original system to

$$Y^T \left(-\Omega^2 M + i\Omega D + K\right) Yq = Y^T f(\Omega) \tag{18}$$

$$Fq = Y^T f(\Omega). \tag{19}$$

F is a diagonal matrix if Y contains eigenvectors of the structure. In this case (17) is a usual transformation to modal coordinates. In recent years, other transformation have been proposed in structural dynamics (e.g. [Wilson et al.82], [Nour-OmidClough84] or [Coutinho et al.87]). A transformation to Lanczos coordinates was shown to be especially promising because it is a load-dependent transformation that is tailored to the respective load case and approximates the influence of higher modes as well. The starting vector y_{L0} that is required for the Lanczos iteration (see e.g. [Coutinho et al.87]) is chosen from

$$Ky_{L0} = \overline{f}, \tag{20}$$

$$\overline{f}_i = \max_{\Omega} \sqrt{\mathrm{Re}(f_i(\Omega))^2 + \mathrm{Im}(f_i(\Omega))^2} \qquad i = 1(1)n. \tag{21}$$

$\overline{f}$ contains an assemblage of the frequency-dependent load $f(\Omega)$ which ensures that every required degree of freedom is excited. If the Lanczos tranformation is performed, for example according to [Coutinho et al.87], a reduced set of equations is attained, which is again diagonal. Thus, both modal and Lanczos transformation yield a reduced diagonal set

$$diag(-\Omega^2 + i\Omega 2\xi_i\omega_i + \omega_i^2)q = Y^T f(\Omega). \tag{22}$$

For optimization purposes the derivatives of (14) are needed. Combining the derivatives of (14) and (16) one has

$$\frac{\partial g_i}{\partial x} = -\frac{1}{a_{i,max}} \frac{\frac{\partial \mathrm{Re}(y_i)}{\partial x}\mathrm{Re}(y_i) + \frac{\partial \mathrm{Im}(y_i)}{\partial x}\mathrm{Im}(y_i)}{a_i}. \tag{23}$$

To compute the derivatives of the complex displacement vector y which are needed in (23), equations (15) and (17) are differentiated:

$$\frac{\partial y}{\partial x} = Y\frac{\partial q}{\partial x} + \frac{\partial Y}{\partial x}q \tag{24}$$

$$diag(-\Omega^2 + i\Omega 2\xi_i\omega_i + \omega_i^2)\frac{\partial q}{\partial x} = \frac{\partial Y^T}{\partial x} f(\Omega) - diag(\frac{\partial \omega_i^2}{\partial x} + i\Omega 2\frac{\partial(\xi_i\omega_i)}{\partial x})q. \tag{25}$$

Thus, computing the gradients of (14) in a reduced system requires the derivative of the transformation matrix Y. If Y contains a set of eigenvectors of the structure, the eigenvector derivatives have to be computed. This can be done by the method originally proposed by [Nelson76] or by some approximation method (see, for example, [Haftka et al.89]) which is usually based on a modal expansion of the eigenvector derivatives.

Numerical experience indicates that the influence of transformation matrix derivatives on the gradients of the frequency response is often negligible. Differentiating (18) and neglecting the derivatives of the damping matrix and of the transformation matrix Y, (24) and (25) are approximated as

$$\frac{\partial \overline{y}}{\partial x} = Y\frac{\partial \overline{q}}{\partial x} \tag{26}$$

$$diag(-\Omega^2 + i\Omega 2\xi_i\omega_i + \omega_i^2)\frac{\partial \overline{q}}{\partial x} = -Y^T \left(-\Omega^2 \frac{\partial M}{\partial x} + \frac{\partial K}{\partial x}\right) Yq. \tag{27}$$

Equations (26) and (27) can be evaluated with Y containing either eigenvectors or some other transformation vectors which, for example, can be obtained by a Lanczos reduction.

EXAMPLE 4 For a cantilever beam a constraint is formulated for the acceleration at the tip of the beam. Different methods for computing the frequency response derivatives are compared. An approximation of the eigenvector derivatives by the method proposed by [Lim et al.87] gives identical results compared to the numerical gradient and the gradient obtained by Nelson's method. An expansion of the eigenvector derivatives in the eigenvectors themselves

$$\frac{\partial Y}{\partial x} = YC^T \tag{28}$$

with C as the matrix of the expansion coefficients, yields very good results for the response gradient, too (figure 6a)). Neglecting the eigenvectors derivatives totally (i.e. using equations (26) and (27)) also gives acceptable results as shown in figure 6b). A difference between the exact and the approximated gradient can be found in the higher frequency range only. ∎

EXAMPLE 5 Using the approximation method for the gradient, a cantilever plate (fig. 7a)) is optimized. Fig. 7b) shows the amplitude response spectra in the initial and final design. The weight history is given in fig. 7c). ∎

133
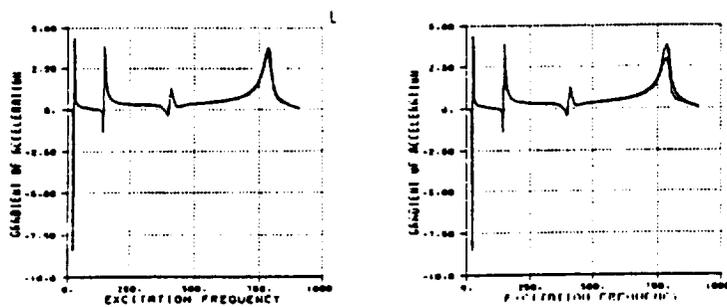
Figure 6: Comparison of exact and approximated frequency response gradients
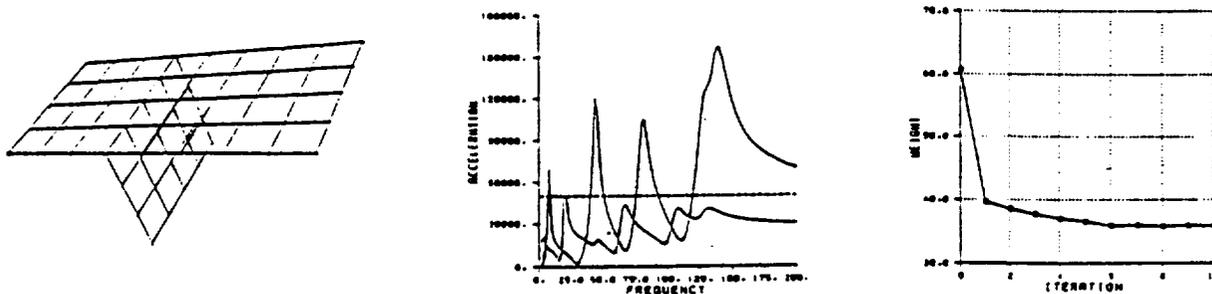
Figure 7: a) cantilever plate b) amplitude response spectra c) weight history

4 Conclusions

Several formulations for eigenfrequency constraints were presented and implemented in the structural optimization package MBB-LAGRANGE. Examples show that the given formulations are very useful in practical applications.

The second topic of the paper addresses frequency response constraints. To save computational time it is necessary to perform a system reduction both in analysis and gradient calculations. The system equations are transformed to either modal or Lanczos coordinates. Neglecting the derivative of the transformation results in equations that are easy and inexpensive to handle. The user may, however, consider exact or approximated eigenvector derivatives in the response gradients as well.

Work on transient response and random response is in progress. Together with the formulations given above LA-GRANGE can be used as a tool for solving a variety of structural optimization problems with constraints from dynamics.

References

[Coutinho et al.87]   Alvaro L.G.A. Coutinho, Luiz Landau, Edison C.P. Lima, and Nelson F.F. Ebecken. The Application of the Lanczos Mode Superpostion Method in Dynamic Analysis of Offshore Structurs. *Computers & Structures*, 25(4) 615–625, 1987.

[Haftka et al.89]   Raphael T. Haftka, Manohar P. Kamat, and Zafer Gürdal. *Elements of Structural Optimization*. Martinus Nijhoff Publishers, The Hague Boston Lancaster, 2, 1989.

[Kneppe89]   Guenther Kneppe. Multicriterion Optimal Layouts of Aircraft and Spacecraft Structures. In *Multicriterion Design Optimization - Procedures and Applications*, H.Eschenauer and J.Koski and A.Osyczka, 1989.

[Lim et al.87]   K.B. Lim, J.L. Junkins, and B.P.Wang. Re-examination of Eigenvector Derivatives. *Journal of Guidance and Control*, 10(6), 1987.

[Nelson76]   Richard B. Nelson. Simplified Calculation of Eigenvector Derivatives. *AIAA Journal*, 14(9), 1976.

[Nour-OmidClough84]   Bahram Nour-Omid and Ray W. Clough. Dynamic Analysis of Structures Using Lanczos Co-Ordinates. *Earthquake Engineering and Structural Dynamics*, 12 565–577, 1984.

[Wilson et al.82]   Edward L. Wilson, Ming-Wu Yuan, and John M. Dickens. Dynamic Analysis By Direct Superposition of Ritz Vectors. *Earthquake Engineering and Structural Dynamics*, 10 813–821, 1982.

DYNAMIC OPTIMIZATION THEORY WITH
MULTIPLE OBJECTIVES

JOHN JONES, JR.
Air Force Institute of Technology

ABSTRACT. Let $V(t)$ be a vector-valued function for $t\epsilon[a,b)$ a real interval. The main purpose of this paper is to establish the existence of an interval $[\alpha,\beta] \subseteq [a,b]$ for which there exists a $t_0\epsilon[\alpha,\beta] \subseteq [a,b]$ such that $V(t_0) = 0$, the zero vector. Use of such information in the dynamic optimization theory with multiple objectives present is needed. Examples of such systems will be given.

INTRODUCTION

Let $V(t)$ be a vector-valued function for $t\epsilon[a,b]$ and denoted by:

$$V(t) = \begin{pmatrix} v_1(t) \\ v_2(t) \\ \vdots \\ v_n(t) \end{pmatrix} \tag{1.1}$$

where each $\{v_i(t), i=1,2,\ldots,n\}$ are twice continuously differentiable real-valued functions of t for $t\epsilon[a,b]$. Associated with $V(t)$ is the following non-self-adjoint matrix vector differential equation:

$$\left[[P(t)]V'(t)\right]' + Q(t)V'(t) + R(t)V(t) = 0 \tag{1.2}$$

where $[P(t)]$, $[Q(t)]$, $[R(t)]$ are n by n matrices of continuous functions of $t\epsilon[a,b]$ and $[P(t)]$ is an n by n matrix of continuously differentiable functions of $t\epsilon[a,b]$.

The main objective of this paper is to establish a sufficient condition for the existence of a $t_0\epsilon[a,b]$ for $V(t)$ a solution of (1.2) such that $V(t_0) = 0$, a zero vector of $V(t)$ for which $t_0\epsilon[\alpha,\beta] \subseteq [a,b]$. Also associated with (1.2) is the following matrix vector differential equation:

$$u''(t) + [A(t)]u'(t) + [D(t)]u(t) = 0 \tag{1.3}$$

where the matrices $[A(t)]$, $[D(t)]$ are continuous functions of a real variable $t\epsilon[a,b]$ such that $u(a) = u(b) = 0$. Such solutions of (1.3) will be called trial functions where $u(t) \neq 0$, for $a < t < b$. Such vector-valued functions $u(t)$ will be utilized in a computational decision process for the existence of a $t_0\epsilon[\alpha,\beta] \subseteq [a,b]$ such that $V(t_0) = 0$.

In section II an example which illustrates the basic theory which is developed in III will be given.

EXAMPLES

Example 1. Let

$$V(t) = \begin{pmatrix} 0 \\ t \end{pmatrix} \quad ; \quad V(o) = 0 \tag{1.4}$$

be a given vector-valued function of t for $t\epsilon[a,b]$ and is a solution of the following matrix vector differential equation

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & -t \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ t \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{1.5}$$

The basic problem of determining a t-interval $[\alpha,\beta] \subseteq [a,b]$ for which $V(t_o) = 0$ where $t_o\epsilon[\alpha,\beta] \subseteq [a,b]$ and $V(t_o) = 0$ will be considered.

Next, let a trial function be given by

$$u(t) = \begin{pmatrix} t^2-t \\ 0 \end{pmatrix} \tag{1.6}$$

which obviously has t-zeros at $t = 0$, $t = 1$ and $u(t) \neq 0$ for $0 < t < 1$. Thus t_o for which $u(t_o) = 0 = u(0)$ and $u(t_o) = 0$ for which $u(1) = 0$. Let $u(t)$ given by (1.6) be a solution of the following matrix vector differential equation

$$\begin{pmatrix} 2 \\ 0 \end{pmatrix} + \begin{pmatrix} 2-4t & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 2t-1 \\ 0 \end{pmatrix} + \begin{pmatrix} 8 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} t^2-t \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{1.7}$$

and $u(t)$ has t_o-zeros at $t_o = 0$, $t_o = 1$.

Let

$$[A(t)] = \begin{pmatrix} 2-4t & 0 \\ 0 & 0 \end{pmatrix} \quad ; \quad [D(t)] = \begin{pmatrix} 8 & 0 \\ 0 & 1 \end{pmatrix} \tag{1.8}$$

for equation (1.7) above.

Then by making use of the following decision-function:

$$E[u,v; \; a,b] \triangleq \frac{1}{2} \left[<Au,u>' \right]^2 - \frac{[<Au,u>]^2}{<v,v>} \cdot \left[<Bv',v> - <v',v'> + <Cv,v> \right]$$

$$= \frac{1}{2} \left[< \begin{pmatrix} 2-4t & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} t^2-t \\ 0 \end{pmatrix}, \begin{pmatrix} t^2-t \\ 0 \end{pmatrix} >' \right]^2$$

$$- \frac{\left[< \begin{pmatrix} 2-4t & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} t^2-t \\ 0 \end{pmatrix}, \begin{pmatrix} t^2-t \\ 0 \end{pmatrix} > \right]^2}{< \begin{pmatrix} 0 \\ t \end{pmatrix}, \begin{pmatrix} 0 \\ t \end{pmatrix} >} \cdot \left[< \begin{pmatrix} 1 & 0 \\ 0 & -t \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ t \end{pmatrix} > \right.$$

$$\left. - < \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} > + < \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ t \end{pmatrix}, \begin{pmatrix} 0 \\ t \end{pmatrix} > \right]$$

$$= \frac{1}{2} \left[< \begin{pmatrix} -2t+6t^2-4t^3 \\ 0 \end{pmatrix}, \begin{pmatrix} t^2-t \\ 0 \end{pmatrix} >' \right]^2 \tag{1.9}$$

$$- \frac{\left[< \begin{pmatrix} -2t+6t^2-4t^3 \\ 0 \end{pmatrix}, \begin{pmatrix} t^2-t \\ 0 \end{pmatrix} > \right]^2}{t^2} \cdot [-t^2-1+t^2]$$

and therefore $V(t)$ has a $t \epsilon [0,1] = [\alpha, \beta] \subseteq [u,b]$ such that $V(t_o) = V(o) = 0$ since

$$\int_o^1 E[u,v; \; 0,1]dt < 0 \tag{1.10}$$

holds.

Example 2. Given the matrix vector differential equation for which the given vector $v(t) = \begin{pmatrix} t^2-t \\ t \end{pmatrix}$ is a solution: (which has $t_o = 0$)

137

$$\begin{pmatrix} 2 \\ 0 \end{pmatrix} + \begin{pmatrix} 2-4t & 0 \\ 0 & -t \end{pmatrix} \begin{pmatrix} 2t-1 \\ 1 \end{pmatrix} + \begin{pmatrix} 8 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} t^2-t \\ t \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

where $B(t) = \begin{pmatrix} 2-4t & 0 \\ 0 & -t \end{pmatrix}$; $C(t) = \begin{pmatrix} 8 & 0 \\ 0 & 1 \end{pmatrix}$

Let the trial vector

$u(t) = \begin{pmatrix} t^2-t \\ t-t^2 \end{pmatrix}$ which has a t_o-zero at $\{t_o = 0,1\}$, $u(o) = 0$; $u(1) = 0$, $u(t) \ddagger 0$

and satisfies the matrix differential equation:

$$\begin{pmatrix} 2 \\ -2 \end{pmatrix} + \begin{pmatrix} 2-4t & 0 \\ 0 & 2-4t \end{pmatrix} \begin{pmatrix} 2t-1 \\ 1-2t \end{pmatrix} + \begin{pmatrix} 8 & 0 \\ 0 & 8 \end{pmatrix} \begin{pmatrix} t^2-t \\ t-t^2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

where

$$u''(t) + A(t)u'(t) + D(t)u(t) = 0$$

and

$$A(t) = \begin{pmatrix} 2-4t & 0 \\ 0 & 2-4t \end{pmatrix} ; \quad D(t) = \begin{pmatrix} 8 & 0 \\ 0 & 8 \end{pmatrix}$$

(Decision Criteria)

$$E[u,v; a,b] \triangleq \frac{1}{2} \left[\langle Au,u \rangle' \right]^2 - \frac{[\langle Au,u \rangle]}{\langle v,v \rangle} \cdot \left[\langle Bv',v \rangle - \langle v',v' \rangle + \langle Cv,v \rangle \right]$$

$$= \frac{1}{2} \left[\left\langle \begin{pmatrix} 2-4t & 0 \\ 0 & 2-4t \end{pmatrix} \begin{pmatrix} t^2-t \\ t-t^2 \end{pmatrix} , \begin{pmatrix} t^2-t \\ t-t^2 \end{pmatrix} \right\rangle' \right]^2$$

$$- \frac{\left[\left\langle \begin{pmatrix} 2-4t & 0 \\ 0 & 2-4t \end{pmatrix} \begin{pmatrix} t^2-t \\ t-t^2 \end{pmatrix} , \begin{pmatrix} t^2-t \\ t-t^2 \end{pmatrix} \right\rangle \right]^2}{\left\langle \begin{pmatrix} t^2-t \\ t \end{pmatrix} , \begin{pmatrix} t^2-t \\ t \end{pmatrix} \right\rangle} \cdot$$

$$\left[\left\langle \begin{pmatrix} 2-4t & 0 \\ 0 & -t \end{pmatrix}\begin{pmatrix} 2t-1 \\ 1 \end{pmatrix}, \begin{pmatrix} t^2-t \\ t \end{pmatrix}\right\rangle \cdot \left\langle \begin{pmatrix} 2t-1 \\ 1 \end{pmatrix}\begin{pmatrix} 2t-1 \\ 1 \end{pmatrix}\right\rangle\right.$$

$$\left. + \left\langle \begin{pmatrix} 8 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} t^2-t \\ t \end{pmatrix}, \begin{pmatrix} t^2-t \\ t \end{pmatrix}\right\rangle\right]$$

$$= \frac{1}{2}\left[\left\langle \begin{pmatrix} 6t^2-2t-4t^3 \\ 2t-6t^2+4t^3 \end{pmatrix}, \begin{pmatrix} t^2-t \\ t-t^2 \end{pmatrix}\right\rangle'\right]^2$$

$$- \frac{\left[\left\langle \begin{pmatrix} 6t^2-2t-4t^3 \\ -6t^2+2t+4t^3 \end{pmatrix}, \begin{pmatrix} t^2-t \\ t-t^2 \end{pmatrix}\right\rangle\right]^2}{[(t^2-t)^2+t^2]}$$

$$\left[\left\langle \begin{pmatrix} 8t-2-8t^2 \\ -t \end{pmatrix}, \begin{pmatrix} t^2-t \\ t \end{pmatrix}\right\rangle - [(2t-1)^2+1]\right.$$

$$\left. + \left\langle \begin{pmatrix} 8t^2-8t \\ t \end{pmatrix}, \begin{pmatrix} t^2-t \\ 1 \end{pmatrix}\right\rangle\right]$$

which guarantees the existence of a t-zero of v(t) for $t\epsilon[0,1]$.

BASIC MATHEMATICAL FORMULATION OF MULTIPLE OBJECTIVE DECISION ANALYSIS MODELS

Multiple-criteria decision making (MCDM) has increased the need to identify and consider simultaneously several objectives, particularly those derived from the study of large-scale dynamical systems. The type of systems considered in this work depend continuously upon time t with respect to the time-varying dynamical system. Mathematical programming is a very useful and flexible framework for multi-objective analysis when the objective and physical constraints of a problem may be expressed as functions of decision variables.

The general problem of single-objective programming is the search for the optimum, i.e., a minimum or maximum of a function of variables constrained by equations or inequalities called constraints. The inequalities may also be differential inequalities or integral inequalities. A single-objective constrained optimization problem can be defined and expressed as below:

$$\max (z(x))$$

139

subject sets of inequalities:

$$
\begin{cases}
g_i(x) \le 0 \; ; \; (i=1,2,\ldots,m) \\
x_j \ge 0 \quad ; \; (j=1,2,\ldots,n)
\end{cases}
$$

where the objective function $z(x)$ and the constraints $g_i(x)$ are defined on an n-dimensional Euclidean vector space of decision variables

$$ x = (x_1, x_2, x_3, \ldots x_n) \epsilon R^n $$

with values in the set of real numbers R. The functions $z(x)$ and $g_i(x)$ can be either linear or nonlinear functions of the decision variables, x_j, where $(j=1,2,3,\ldots,n)$. The feasible region is denoted by $X = \{x : x \epsilon R^n, \, g_i(x) \le 0, \, x_j \ge 0, \, \forall \, i,j\}$. The optimization problem is to find an element $x^* \epsilon X$ of the feasible region X which will give a minimum value for $z(x)$, i.e., $\max z(x) = z(x^*)$. If the functions $z(x)$ and $g_i(x)$ are both linear, then the optimization problem is called a linear programming problem. If either $z(x)$ and $g_i(x)$ are nonlinear, then such a problem is called a nonlinear programming problem.

A single-objective programming problem consists of optimizing one objective function subject to a constraint set. A multi-objective programming problem is characterized by a p-dimensional vector $z(x)$ of objective functions:

$$ z(x) = (z_1(x), z_2(x), \ldots, z_p(x))^t, \; t = \text{transpose} $$

and a feasible region $X = \{x : x \epsilon R^n, \, g_i(x) \le 0, \, x_j \ge 0, \, \forall \, i,j\}$, but instead of seeking a single-optimal solution, a set of nondominated solutions is desired to be found. This set of nondominated solutions is a subject of the feasible region $X = \{x : x \epsilon R^n, \, g_i(x) \le 0, \, x_j \ge 0, \, \forall \, i,j\}$. For the nondominated set of solutions, each solution outside the set, but within the feasible region X, there is a nondominated solution for which all objective functions are unchanged or improved where also at least one is strictly improved. In general, one can't optimize a priori a vector of objective functions. One first attempts to identify the set of nondominated solutions within the feasible region X in the consideration of multi-objective problems.

One may formulate the problem as follows:

$$ \text{max-dominate } z(x) = [z_1(x), z_2(x), \ldots, z_p(x)] $$

subject to $x \epsilon X$ the feasible region, where the object is to search and identify the set of nondominated solutions, i.e., the set of solutions that dominate the other solutions in X. Decision functions are used in this area for time-varying problems developed by J. Jones, Jr. and the detailed mathematical proofs will appear elsewhere due to page limitation of this manuscript. Parallel processing and large computers are used to carry out the mathematical modelling and computer simulation of large dynamical time-varying systems (both linear and nonlinear).

FUZZY COMPROMISE: AN EFFECTIVE WAY TO SOLVE HIERARCHICAL DESIGN PROBLEMS

J.K. ALLEN[1], R.S. KRISHNAMACHARI[2], J. MASETTA[3] , D. PEARCE[3], D. RIGBY[3]
AND
F. MISTREE[4]

In this paper, we present a method for modeling design problems using a compromise Decision Support Problem (DSP) incorporating the principles embodied in fuzzy set theory. Specifically, the fuzzy compromise Decision Support Problem is used to study hierarchical design problems. This approach has the advantage that although the system modelled has an element of uncertainty associated with it, the solution obtained is crisp and precise. The efficacy of incorporating fuzzy sets into the solution process is discussed in the context of results obtained for a portal frame.

1   MODELING HIERARCHICAL PROBLEMS AS COMPROMISE DSPs

Many real-world engineering systems are too complex to be designed as single systems. In the early stages of design it is essential to have a method for partitioning and/or decomposing design problems into subsystems which then may be designed concurrently. The nature of the problem itself and the method chosen for analyzing the subsystems have dramatic effects on the efficiency and effectiveness of the design process as a whole, and by extension, on its cost. Design through repeated iteration is costly, time consuming, and may require endless iterations to adjust previously "designed" subsystems to take into account new information. On the other hand, completely simultaneous design (except in the case of variant design) is also impossible. The general formulation of a hierarchically decomposed problem is shown in Figure 1. Methods for hierarchical decomposition have been proposed and tested successfully in limited situations; Sobieski [1-3], Kuppuraju et al.[4], Shupe et al.[5], Wrenn and Dovi [6], and Padula et al. [7]. There is, however, a major limitation to these methods, they require precisely defined information and relationships between subsystems, and therefore they are impractical for use in the early stages of project initiation. We assert that a procedure that incorporates fuzzy set theory would overcome this limitation.

An abstract system made up of a parent system and three subsystems is illustrated in Figure 1a. The subsystems interact with each other through *lateral interactions* and with the parent system through *vertical interactions*. A physical representation of the abstract system (shown in Figure 1a) is a portal frame, Figure 1b. It represents a simple hierarchical system; the frame being the parent system and the I-beams three subsystems. The objective is to minimize the overall mass of the frame while it is subjected to static loads P and M. The system is subject to certain constraints covering normal stress, bending stress, shear stress and buckling in each member. There are two types of design variables; one type for the parent system and another type for the subsystems. The parent system design variables (A and I) are each member's cross sectional area and moment of inertia. Each subsystem (I-beam) has six design variables, namely, b_1, b_2, h, t_1, t_2, t_3. The vertical interactions, V_i, that occur between the parent system and each of its subsystems, as well as the interactions that occur between each subsystem, L_{ij}, are shown in Figure 1b. The lateral interactions necessitate the inclusion of constraints that match the subsystem variables to their counterparts in the other subsystems. The vertical interactions necessitate the inclusion of constraints that match the parent system design variables (A and I) and the subsystem variables (b, t, h). This "matching" is modeled mathematically by system goals in the compromise DSP. An interaction system goal constrains one subsystem variable to be equal to its counterpart in the other subsystem. For example,

---

1   President, JANCO Research, Inc., 4501 University Oaks Boulevard, Houston, Texas 77004.
2   Graduate Research Assistant, Department of Mechanical Engineering, University of Houston, Houston, Texas 77204.
3   Graduate Student, Department of Mechanical Engineering, University of Houston, Houston, Texas 77204.
4   Professor, Department of Mechanical Engineering, University of Houston, Houston, Texas 77204.
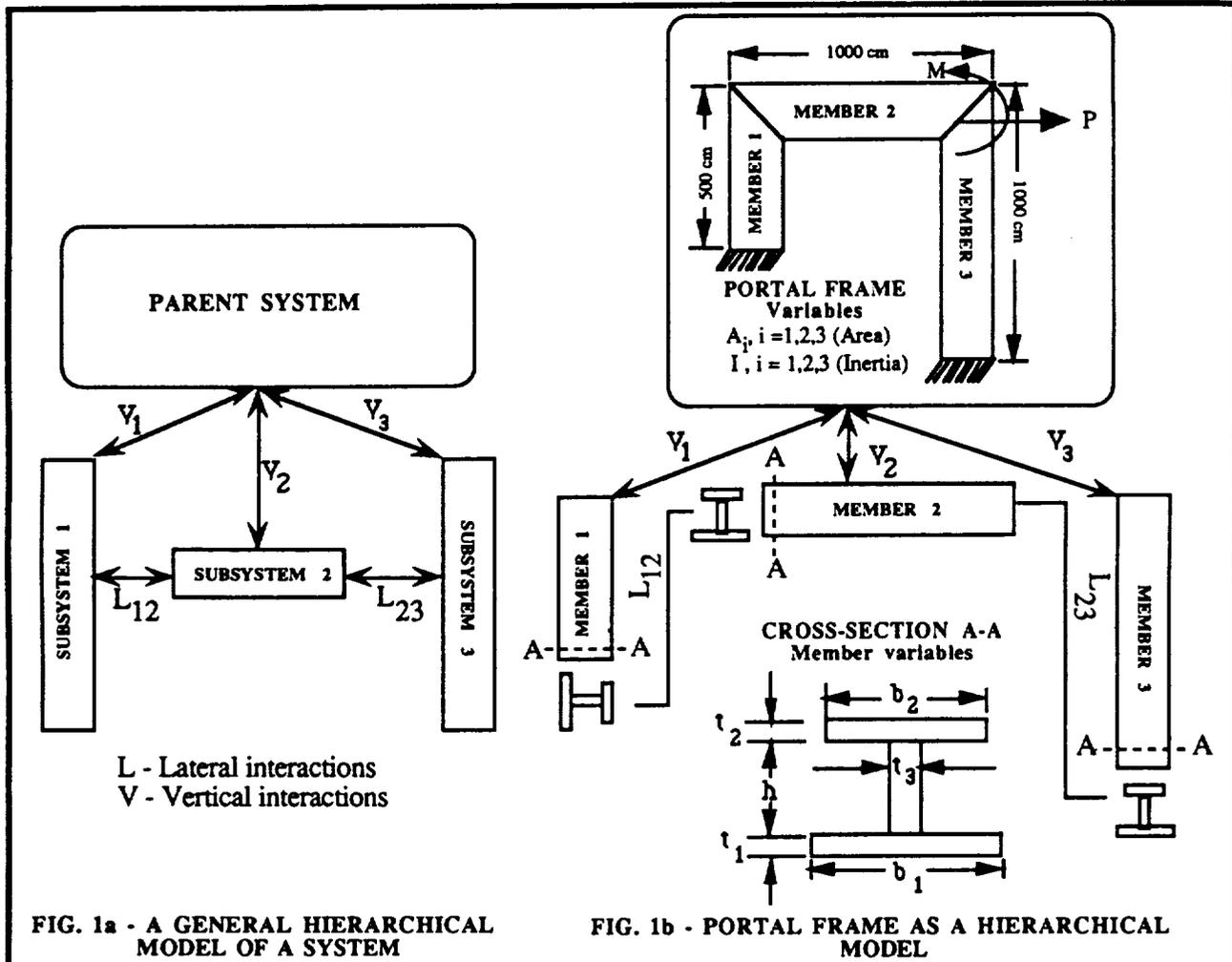
FIG. 1a - A GENERAL HIERARCHICAL MODEL OF A SYSTEM

FIG. 1b - PORTAL FRAME AS A HIERARCHICAL MODEL

GIVEN
Geometry, material properties, loads, etc.
X^0 - the starting solution
i = 1,2,3 - member number
j = 1 and 2 - left and right end of member, respectively

FIND
$X = \{A_i, I_i, b_{i1}, b_{i2}, t_{i1}, t_{i2}, h_i\}$

SATISFY
System Constraints: **Frame**
- combined stress constraints
$S(A_i, I_i) \le S_{max}$

System Constraints: **For each member**
- combined stress in the top flange
$\sigma(b,t,h)_{ij} \le \sigma_a$
- combined stress in the bottom flange
$\sigma(b,t,h)_{ij} \le \sigma_a$
- shear stress constraint
$\tau(b,t,h)_{ij} \le \tau_a$

- flange buckling constraint (top & bottom)
$|\sigma(b,t,h)_i| \le \sigma_{ab}(b_{i1}, t_{i1})$
$|\tau(b,t,h)_i| \le \tau_{ab}(b_{i1}, t_{i1})$

Overall System Goal
- Mass Goal: minimize mass of system

Other System Goals
- Vertical Interaction Goals: V_i (six goals)

$A_i + d^-_{i+1} - d^+_{i+1} = A(b, t, h)_i$

$I_i + d^-_{i+4} - d^+_{i+4} = I(b, t, h)_i$

- Lateral Interaction Goals: L_{12} and L_{23}
(twelve goals)

$L_1(b, t, h) + d^-_n - d^+_n = L_2(b, t, h)$

$L_2(b, t, h) + d^-_n - d^+_n = L_3(b, t, h)$

Bounds on system and deviation variables

MINIMIZE
The mass of the system
The deviation between the system variable
in one member and its counterpart in the other.

FIG. 1c - MATHEMATICAL MODEL OF THE HIERCHICAL DSP FOR THE PORTAL FRAME [5]

FIGURE 1 - THE HIERARCHICAL PROBLEM AND THE COMPROMISE DSP

in Figure 1b, the individual dimensions of the center beam (subsystem 2) of the portal frame should match those of the beams (subsystems 1 and 3) on either side. Thus, lateral interaction equality constraints are created to handle this, for example,

$$(b_1)_1 + d^- - d^+ = (b_2)_1$$

where $(b_1)_1$ and $(b_2)_1$ are the width of the bottom flanges of subsystems 1 and 2, respectively. The deviation from this equality is measured by the system goal's deviation variables (underachievement, d^- and overachievement, d^+)[1]. Similarly, the interaction between the parent system and subsystem 1 is written as

$$A_1 + d^- - d^+ = A(b,t,h)_1$$

where A_1 is the cross sectional area of member 1 (parent system) and $A(b,t,h)_1$ is the cross sectional area of member 1 as a function of the subsystem variables. The interaction system goal provides an effective and efficient approach for maintaining the interactions between the parent system and its subsystems and between individual subsystems. Further details are provided in [5].

The crisp formulation, in succinct notation, of the compromise DSP for the portal frame is presented in Figure 1c. The compromise DSP has been discussed in detail elsewhere (for example, see ref [10]) and will not be repeated here). Given the geometry, loads, material properties and starting values for the variables X^0 our intention is to find that set of variables X that minimizes the mass (modeled as volume in the formulation [5]) and satisfies system constraints and goals, and bounds on the variables. The system constraints include stress constraints on the parent system and stress and buckling constraints on the subsystems. The system goals include vertical and lateral interactions. Bounds are placed on both the system and deviation variables. The objective, unlike traditional optimization formulations, is in terms of the deviation variables associated with the goals. The stresses are calculated using SAP IV and the compromise DSP is solved using DSIDES [10]. Shupe and coworkers showed the efficacy of using compromise DSPs in modeling and solving hierarchical design problems [5].
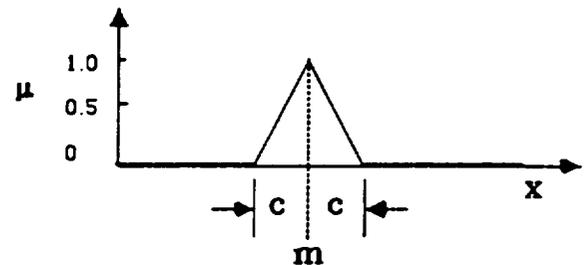
FIGURE 2 - A ONE DIMENSIONAL LINEAR MEMBERSHIP FUNCTION

2 FUZZY SETS IN DESIGN

What is a fuzzy number? Using fuzzy set theory, uncertainty in any variable is modeled by assuming that it is represented by a main value, m, surrounded by a cloud of fuzziness (uncertainty) [8, 9] whose shape is specified by a membership function, μ. The exact shape of the membership function must conform to some conventions and numerical considerations: (a) its only minima are at the end points of the interval and these must have a zero membership, and (b) its only maximum should have membership one at the most likely value within the fuzzy set. A simple function that satisfies these guidelines is the linear membership function which is illustrated in Figure 2. In this case, the most likely value of the set is the main value, m, and the distribution of membership decreases linearly and symmetrically until it equals zero at (m+c) and (m-c). Hence, linear membership functions are completely specified by the value of m and the bandwidth of fuzziness, c.

What is the difference between fuzzy set theory and statistics? The analysis of Gaussian distributions in probability and statistics is based on the rigorous mathematical foundation of measure theory. It is inappropriate to apply identical manipulations to fuzzy sets - even the very basic definitions of addition and multiplication of sets are different in the two cases.

The general formulation of a fuzzy compromise DSP is given below.

---

1   The system and deviation variables in a compromise DSP are always non-negative. To effect solution, one of the following three conditions must hold, namely; [$(d_k^- = 0)$ and $(d_k^+ = 0)$] or [$(d_k^- = 0)$ and $(d_k^+ > 0)$] or [$(d_k^- > 0)$ and $(d_k^+ = 0)$]. This requirement is modeled by: [$(d_k^- \cdot d_k^+) = 0$].

GIVEN

- An *alternative (a starting design):* $\underline{X}^0 = \{X_i^0 \mid i = 1, \dots, M\}$.
- Estimated *fuzzifiers* for constraints, $\underline{c}_{const.} = \{c_j\}$, and performance goals, $\underline{c}_{goals} = \{c_p\}$.
  $j = 1, \dots, J$ and $p = 1, \dots, P$.
- Upper and lower *bounds* on the system variables.

FIND

- The values of *system variables* : $\underline{X} = \{X_i \mid i = 1, \dots, M\}$.
- *Deviation variables* d_k^+ and d_k^+ are a measure of the deviation of the system being designed from the goals. $k = 1, \dots, K$ where K is the number of goals. In the fuzzy formulation, there are F goals associated with the grade(s) of system compatibility, G goals associated with goal satisfaction, and P performance goals $(F + G + P = K)$.

SATISFY

- *Fuzzy system constraints (capability, C_i , meets demand, D_i)*
  Constraints are a function of the system variables, $\underline{X}$, and constants, A_j. The fuzzy form of these constants is represented as $A_j(1-c_jH_j)$.

  $$[C_j(A_j(1-c_jH_j), \underline{X})] \, / \, [D_j(A_j(1-c_jH_j), \underline{X})] \;\geq\; 1 \; . \qquad j = 1, \dots, J.$$

- *Fuzzy system goals*
  - Performance goals (performance meets target). Performance goals are a function of the system variables, $\underline{X}$, and constants, A_p. The fuzzy form of these constants is represented as $A_p(1-c_pH_p)$.

    $$[P_p(A_p(1-c_pH_p), \underline{X})] \, / \, [T_p(A_p(1-c_pH_p), \underline{X})] + d_p^- - d_p^+ \;=\; 1 \; . \quad p = 1, \dots, P.$$

  - Maximize the grade of system compatibility. (Maximize the possibility of satisfying the fuzzy constraints.)

    $$H_j + d_j^- - d_j^+ = 1. \qquad j = 1, \dots, J.$$

  - Maximize the degree of goal satisfaction. (Maximize the possibility of satisfying the fuzzy goals.)

    $$H_p + d_p^- - d_p^+ = 1. \quad p = 1, \dots, P.$$

- *Bounds*
  - On system variables.

    $$X_i^{min} \leq X_i \leq X_i^{max} \; . \qquad\qquad i = 1, \dots, M.$$

  - On the possibility distribution variables associated with the constraints and goals,

    $$(0 \leq H_j, H_p, H_g, H_f \leq 1) \; . \quad j = 1, \dots, J; \; p = 1, \dots, P; \; g = 1, \dots, G; \; f = 1, \dots, F.$$

MINIMIZE

- *Preemptive deviation function through lexicographic ordering*

  $$Z = [(d_1^- + d_1^+), \dots, (d_k^- + d_k^+), \dots, (d_K^- + d_K^+)] \; .$$

Although fuzziness is introduced into the formulation of the compromise DSP, the design parameters constituting the solution are crisp, i.e., not fuzzy. As the extent of the fuzziness is decreased, $(c \to 0)$, the solution, $\{\underline{X}\}_{fuzzy}$, of the fuzzy compromise DSP approaches the crisp (non-fuzzy) solution, $\{\underline{X}\}_{crisp}$, in the limit.

What are some of the advantages of using a fuzzy compromise DSP?
- Some phenomena, particularly in original design, may be more accurately modeled by fuzzy relationships than by crisp, precise ones.
- A fuzzy formulation particularly useful because it can be used over a relatively large portion of the design time-line. As design proceeds, information about the object being designed becomes more and more certain (less and less uncertain). As the certainty increases a designer is required to merely decrease the value of the fuzzifier, c (the fuzzy formulation reduces to the crisp formulation when c=0).
- Notions such as - "as much as possible" or "approximately 10,000" can be modeled. These are more likely to occur in the early stages of project initiation.
- We believe that the fuzzy compromise DSP provides an added flexibility to the problem's mathematical structure which is very useful in modeling large-scale systems problems. This is particularly so when subsystems are being designed by different groups and must be integrated into a system:
  - In practice, at a point on the time-line, the degree of certainty associated with different subsystems will vary. A fuzzy formulation allows a designer to account for these variations in obtaining a solution for the system.
  - In a fuzzy formulation it is possible to permit some constraints to be "somewhat violated" and then to "instruct" other subsystems to compensate appropriately.
- We believe that the fuzzy compromise DSP provides an ideal way for modeling hierarchical problems. Shupe et al. [5] and Krishnamachari et al. [12] have demonstrated the efficacy of solving these problems using crisp and fuzzy formulations, respectively.

3.0 IMPLEMENTATION, VALIDATION, INSIGHT AND FUTURE WORK

Implementation
To ensure that the DSP template is correct for both the crisp and fuzzy formulations and to gain an insight into the workings of the model four cases of the portal frame were run:
> *Regular* - involves minimizing the system mass only while allowing the geometry of the portal frame to take on any satisficing values.
> *Lateral* - above plus the goal that forces the equivalence of the beam geometries.
> *Vertical* - above plus the goal that forces the equivalence of the beam areas and moment of inertias between the subsystems and the parent system.
> *Comprehensive* - makes use of the goals from the regular formulation, the lateral formulation and the vertical formulation, simultaneously.

Mass is an indirect measure of economic efficiency whereas the interactions are secondary (strength, buckling, etc. being primary) measures of technical efficiency. Hence, in the preceding a preference is shown towards minimizing mass at the expense of the interactions. The solution scheme is illustrated in Figure 3. By way of example a plot of the design history of the volume for three different starting solutions is shown in Figure 4.

Validation
Both the crisp and fuzzy formulations have been exercised with different initial starting solutions to establish the global nature of the solution and to ascertain the effect of infeasible starting solutions on the final design.
> *Establishing the crisp formulation:* The crisp formulation of the DSP proposed by Shupe et al. [5] forms the basis of this study. The volume obtained by Krishnamachari et al. [12] for the lateral and comprehensive cases is within 5 % of the results obtained by Shupe and coworkers and the results for the regular and vertical interaction cases are within 10 %. The difference is attributed to the pseudo-preemptive approach used by Shupe in 1985 and the true preemptive approach used by Krishnamachari and coworkers in 1990.
> *Effect of different starting solutions:* The starting solution did not have much effect on the convergence. All the runs took almost the same number of cycles for convergence for both the crisp and fuzzy formulations. This has been verified for fuzziness varying from 5 to 30 %.
> *Does the fuzzy model behave as expected?* Both formulations have been extensively exercised and the answer is yes.
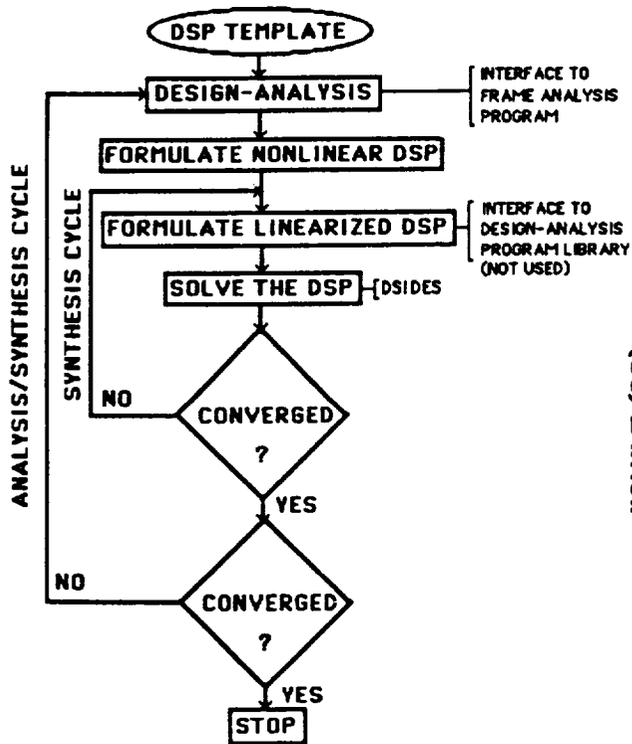
FIGURE 3 - SOLUTION SCHEME   FIGURE 4 - THREE STARTING SOLUTIONS
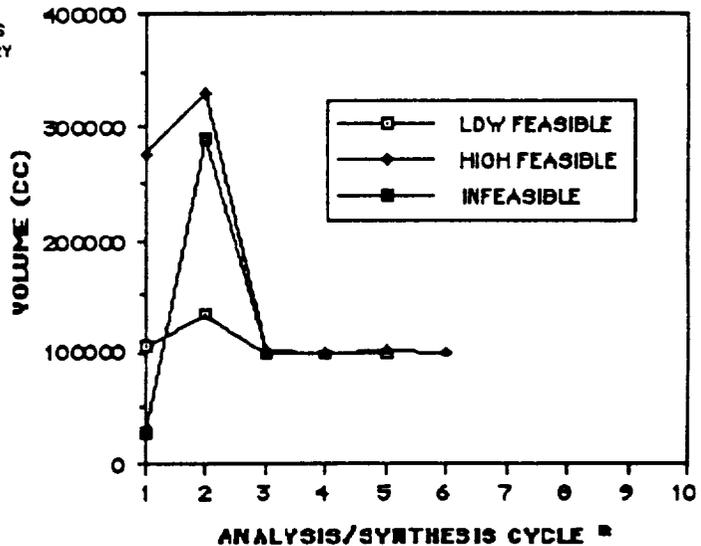
Observations

Using the fuzzy DSP increases the swiftness with which a solution is obtained: The convergence of the fuzzy DSP of the portal frame problem is considerably faster than the corresponding crisp formulation. This rate increases as the fuzziness value is increases. The time for convergence decreased from approximately 12 minutes for the crisp model to 7 minutes for the comprehensive case with a fuzziness of 30 % on the stress limit.

The efficacy of using the fuzzy DSP in the early stages of project initiation: Increasing the fuzziness decreases the effect of an initial infeasible design on the number of iterations needed for convergence. This observation is of particular importance in the early stages of project initiation when little is known of the system and the uncertainties associated with the information are high. Increasing fuzziness increases the speed of convergence and is useful for negating the deleterious effect of an infeasible starting solution on the solution process. The fuzzy model is particularly appropriate when a designer is interested in obtaining a satisficing, approximate design rather than obtaining a design that is accurate and optimal.

The efficacy of using the fuzzy DSP over a range of the design time-line: In practice, uncertainties should decrease as one proceeds along a design time-line. The solution of the fuzzy DSP with c=0 (i.e., no fuzziness) is within 1% of the solution obtained using the crisp formulation. This is indicative that the same model can be used over a wide range of the design time-line.

The utility of the fuzzy DSP to a designer: Post-solution analysis of the uncertainty value 'H' for the constraints that are fuzzified can be used by a designer to determine which factors are of importance in improving the quality of the solution and thence directing his/her efforts accordingly. Fuzziness can be used in two ways, namely, model uncertainties and to study the effect of certain parameters on the solution. The latter allows a designer to pin-point a part (within a subsystem) for which it is important to get further information, i.e., to reduce the uncertainty. This insight should make it possible to direct resources towards activities that provide the "biggest bang for the bucks".

The efficacy of using the fuzzy DSP to model hierarchical problems: Shupe et al. [5] have shown the efficacy of solving hierarchical problems via the compromise DSP. This observation is further reinforced by Krishnamachari et al. [12]. The use of the fuzzy DSP does not result in a better design

146

per se but provides a means for recombining the subsystems into a system when design is in progress. Further, there are strong indications that the use of the fuzzy DSP could result in an increase of the efficiency of the design process.

Future

The portal frame test case represents much more than a structural optimization problem. The same philosophy can be used to model other types of projects in which different departments or groups interact to achieve their goal. The fuzzy compromise DSP can be used to model interactions between systems even though the nature of these interactions may be unclear or fuzzy. This is the case in concurrent engineering. Based on our experience, we believe that investigating the efficacy of using the fuzzy compromise DSP in concurrent engineering is warranted.

ACKNOWLEDGEMENTS

REFERENCES

1   J. Sobieszczanski-Sobieski, *A Linear Decomposition Method for Large Optimization Problems - Blueprint for Development*, NASA TM 83248, Feb. 1982.

2   J. Sobieski, B.B. James and A.R. Dovi, *Structural Optimization by Multilevel Decomposition*. AIAA J, 23, pp. 1775-1782, 1985.

3   J. Sobieszczanski-Sobieski, B.B. James and M.F. Riley, *Structural Sizing by Generalized, Multilevel Optimization*, AIAA J, Vol 25, No. 1, p. 139, 1987.

4   N. Kuppuraju, S. Gannesan, F. Mistree and J.S. Sobieski, *Hierarchical Decision Making in System Design*, Engineering Optimization, Vol. 8, pp. 223-252, 1985.

5   J.A. Shupe, F. Mistree and J.S. Sobieski, *Compromise: An Effective Approach for Design of Hierarchical Structural Systems*, Computers and Structures, Vol. 26, No. 6, pp. 1027-1037, 1987.

6   G.A. Wrenn and A.R. Dovi, *Multilevel Decomposition Approach to the Preliminary Sizing of a Transport Aircraft Wing*, AIAA/ASME/ASCE/AHS, 28th Structures, Dynamics and Materials Conference, Monterey, CA, Paper No. 87-0714-CP, 1987.

7   S.L. Padula, C. Sandridge, R.T. Haftka and J.L. Walsh, *Demonstration of Decomposition and Optimization in the Design of Experimental Space Systems*, Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, Sept. 1988.

8   R.R. Yager, S. Ovchinnikov, R.M. Tong and H.T. Nguyen, Editors, *Fuzzy Sets and Applications: Selected Paper by L.A. Zadeh*, John Wiley and Sons Publishers, New York, NY, 1987.

9   A. Kandel, *Fuzzy Mathematical Techniques with Applications*, Addison-Wesley Publishing Company, Reading, MA, 1986.

10 F. Mistree, S. Marinopoulos, D. Jackson and J.A. Shupe, *The Design of Aircraft using the Decision Support Problem Technique*, NASA Contractor Report, CR 4134, April 1988.

11 Q-J. Zhou, *The Compromise Decision Support Problem: A Fuzzy Formulation*, M.S. Thesis, Department of Mechanical Engineering, University of Houston, May 1988.

12 R.S. Krishnamachari, J.P. Masetta, D. Pearce and D. Rigby, *Hierarchical Design Under Uncertainty*, Systems Design Laboratory Report, Department of Mechanical Engineering, University of Houston, June 1990.

147

AIRFRAME STRUCTURAL OPTIMIZATION FOR MAXIMUM FATIGUE LIFE

By

Dr. D.P. Schrage and A.K. Sareen
Georgia Institute of Technology

ABSTRACT

A methodology is outlined for optimization of airframe structures under dynamic constraints to maximize service life of specified fatigue-critical components. For practical airframe structures, this methodology describes the development of sensitivity analysis and computational procedures for constraints on the steady-state dynamic response displacements and stresses Strain energy consideration is used for selection of structural members for modification. Development of a design model and its relation to an analysis model, as well as ways to reduce the dimensionality of the problem via approximation concepts is described. This methodology is demonstrated using an elastic stick model for the MH-53J helicopter to show service life improvements of the hinge fold region.

INTRODUCTION

Excessive vibrations degrade the service life as well as the ride qualities of helicopters. Studies on vibration reduction by optimization of rotorcraft structures are underway at NASA Langley as a part of an ongoing NASA/Industry rotorcraft structural dynamics program. The objective of these studies is to develop practical computational procedures for structural optimization of airframes subject to steady-state vibration response constraints (Reference 1). Efforts are also underway at developing an integrated, multidisciplinary, optimization-based approach for rotorcraft design (Reference 2). One of the objectives of this NASA/Army Aerostructures Directorate research program is to establish a procedure to include airframe dynamic effects in rotorcraft system dynamic optimization.

While analysis capability for vibratory response has been pursued for quite some time, the vibration reduction in existing helicopters has, for the most part, been achieved through add-on vibration control devices Such after-the-fact structural alterations imply additional weight penalty and are often not effective. These problems can be alleviated if the helicopter designers rely on analysis during design in their efforts to limit vibrations. This will require the development of advanced design analysis methodologies and attendent computational procedures which adequately take vibration requirements into account (Reference 3). This paper outlines an airframe structural design methodology aimed at airframe dynamic structural modification to reduce vibratory response, thereby increasing the service life of fatigue-critical components and producing a better vibration environment for crew, passengers, and equipment.

METHODOLOGY

Airframe structural optimization involves structural analysis capability to be coupled with the optimizer which modifies the design variables so that structural responses meet user-prescribed criteria. For practical helicopter structures, the structural responses are computed using standard structural
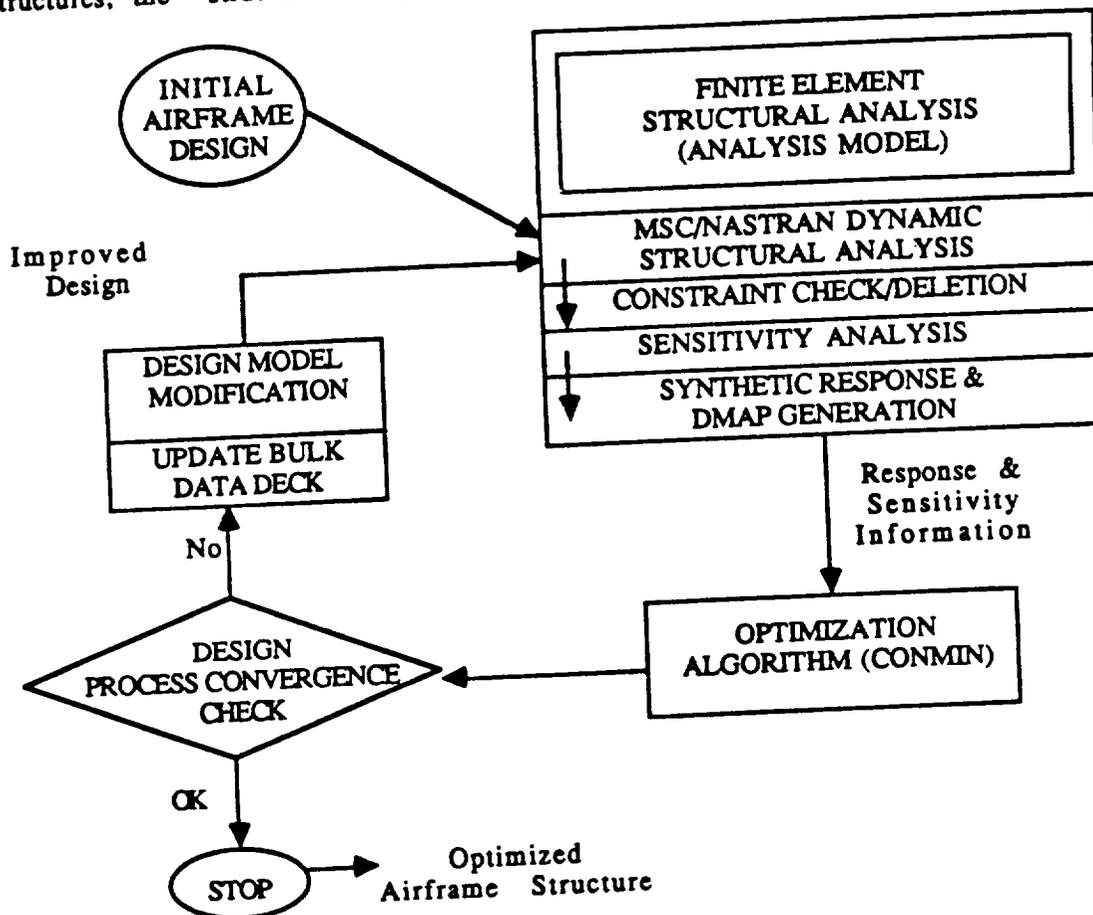
Figure 1. General Airframe Structural Optimization Methodology

analysis tools such as MSC/NASTRAN applied to specifically constructed finite element analysis models. Figure 1 shows the basic airframe structural design optimization process.

Initial airframe design is in the form of a complete finite element analysis model along with the description of rotor-induced loads. Figure 2 shows the finite element model of the MH-53J helicopter which has been subjected to static and dynamic analysis, and the design model is being constructed Once the static analysis has been performed and static-to-dynamic model conversion is completed, the dynamic model is analyzed using MSC/NASTRAN. However, there are obstacles to the implementation of efficient mathematical programming based structural synthesis of the complete finite element model due to

149
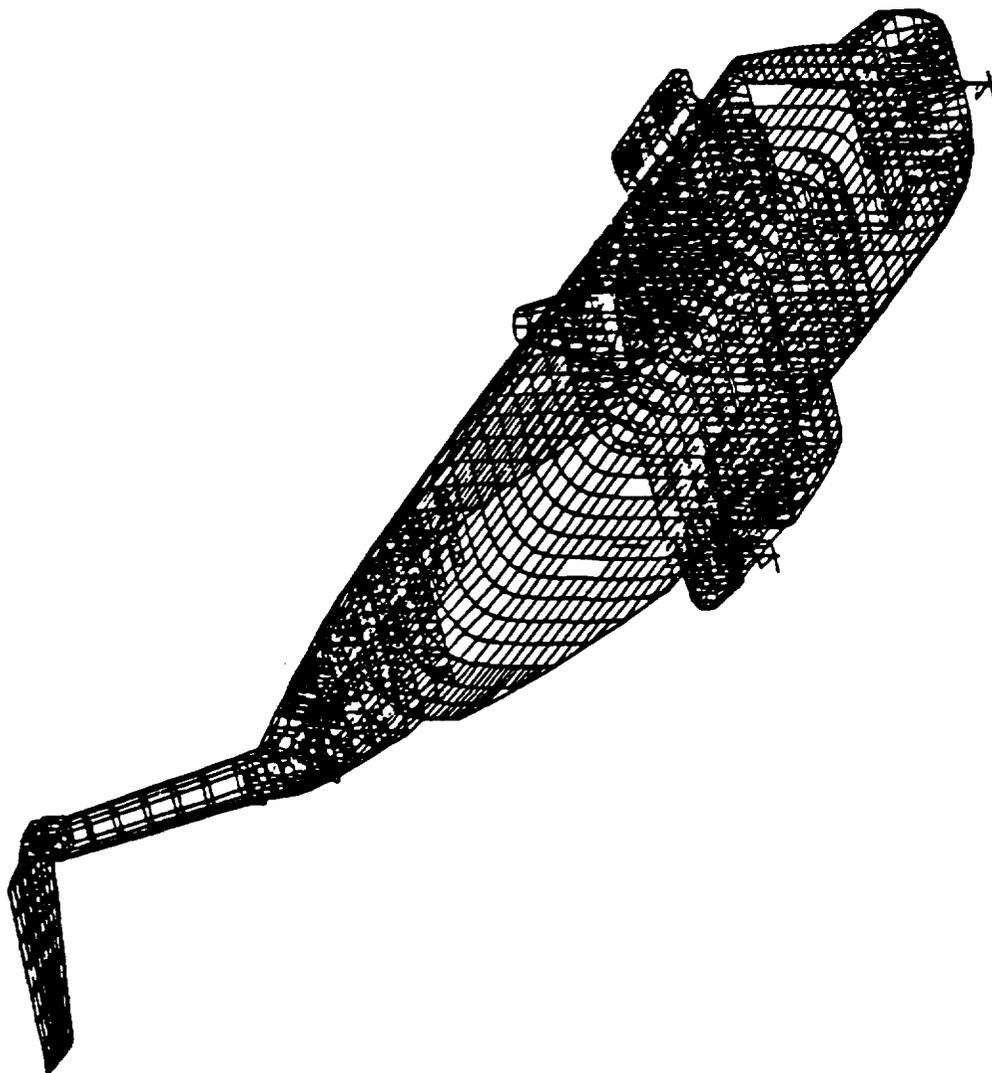
Figure 2. Finite Element Model of the MH-53J Helicopter

i)   large number of design variables
ii)   large number of inequality constraints
iii)   many inequality constraints are computationally
burdensome implicit functions of the design variables.

To overcome the efficiency barrier in structural optimization problems with mathematical programming approaches, approximation concepts can be used. The basic idea is to transform the problem statement involving a large number of implicit functions to a sequence of a relatively small number of approximate problems that incorporate only the explicit functions which are easy to evaluate (Reference 4). The size of the problem is dictated by number of design variables or the number. of constraints. The number of design variables is reduced by linear transformation called design variable linking. The number of constraints is reduced by temporarily deleting constraints which were sufficiently feasible and not likely to become binding for moderate changes of the design variables. Graphically, the approximation concepts methodology is depicted in Figure 3.
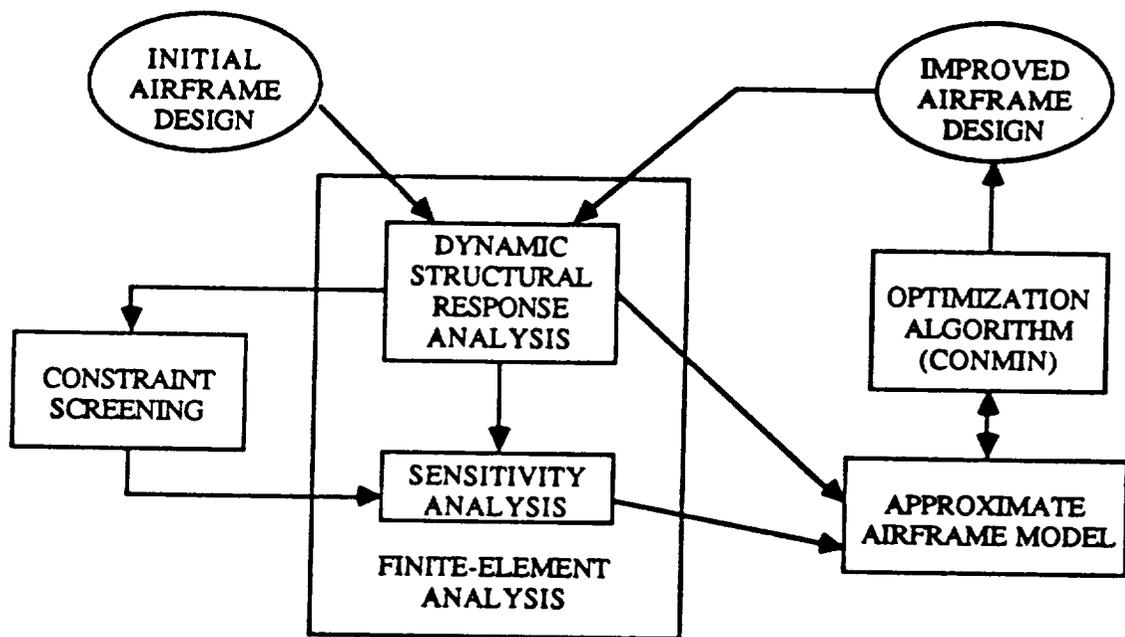
Figure 3.   Approximation Concepts Methodology
(Source: Reference 5)

For the approximate model, the analysis computes the various structural responses, such as stresses, displacements, natural frequencies, etc. together with their sensitivities with respect to a specified set of design model parameters. These response data are fed into the optimization program to propose an improved design. Based on the new design, the analysis model is modified and a new iteration cycle starts.

The use of the design optimization capability for practical airframe structures is relatively simple and the key concept in applying the

optimization capability is to make a "design" model in addition to the available "analysis" model. A design model is a representation of a design optimization problem statement in terms of design variables and structural responses. It is closely related to the corresponding analysis model. To make the best use of optimization capability, the user needs to have a good knowledge of the hardware requirements and design criteria, and then create a design model based either on the proposed structure or the preliminary drawings. The MH-53J design model for optimization is built based on the finite element analysis model of Figure 2.
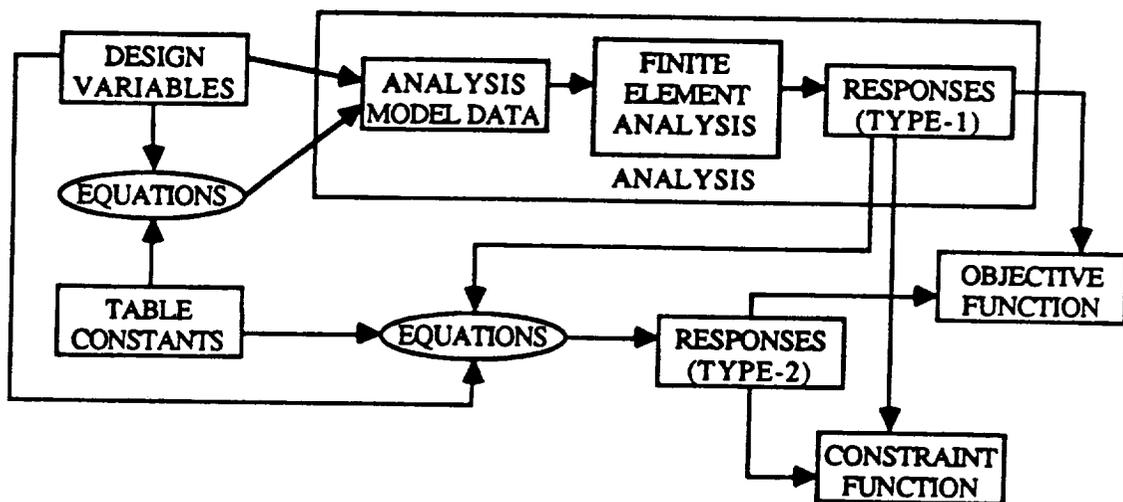
Figure 4. Basic Design Model Conceptual Structure
(Source: Reference 5)

In general, the description of a design model must contain the following four components in addition to the associated analysis model description (Reference 5):

i)   Definition of design variables which are allowed to be modified.
ii)   Description of relations between the analysis model variables and the design variables.
iii)   Description of a measure of design based on the responses calculated by the analyses.
iv)   Description of the design criteria based on the responses calculated by the analysis, in the form of inequality constraints.

Conceptually, the design model description implemented in MSC/NASTRAN may be visualized as shown in Figure 4. The elements in the analysis box represent the conventional MSC/NASTRAN analysis capabilities. The design variables are defined as separate entities from any of the parameters which describe the analysis model. These variables could be normalized area or moment of inertias of the sections. Subsequently their relationships to the analysis model data is defined directly and or through the user supplied equations. In the Figure 4, Type-1 (or direct) responses are those analysis results computed directly by finite element analyses. These are available for printing or for being read by other modules. The objective and constraint functions are formed either directly from the Type-1 responses or

152

from the Type-2 responses which are defined as user-supplied equations in terms of the design variables and the Type-1 responses.

The selection of appropriate design variables is one of the most improtant decisions in creating a design model. It is highly desirable that the design variables selected have an appreciable influence on the objective and/or the constraint functions. Another factor to keep in mind is that each design variable should be directly related to physical significant quantity such as dimension of a part, so that the designer can modify the drawing or the hardware based on the proposed design. The total number of design variables is limited only by the computational resources (i.e. memory size, secondary storage size, etc.). The optimization process is efficient for a reduced number of the design variables. The design model is under development at the writing of this paper, and therefore, not presented here.

MODIFIED METHODOLOGY

The above mentioned generalized methodology, depicted in Figure 1, was modified taking into account the specific efforts undertaken at NASA Langley Research Center in developing the DYNOPT program for tuning frequencies of helicopter airframe structures under dynamic displacement constraints. Ongoing research on helicopter optimization for vibration reduction at Langley Research Center has resulted in the development of computational procedures for optimization of practical airframe structures under dynamic constraints (Reference 6). As a part of these studies, sensitivity analysis procedures for constraints on the steady-state dynamic response displacement of the Bell AH-1G helicopter airframe were developed, under rotor-induced loads. Research work in this regard involved development of a solution sequence based on direct matrix abstraction program (DMAP) of MSC/NASTRAN to compute the sensitivity coefficients for the dynamic response constraints. The sensitivity results from the application of the solution seequence to an elastic line model of a helicopter airframe structure are discussed in Reference 1.

In the research study of Reference 6, a computational procedure based on the nonlinear programming approach of optimization was developed which incorporates the dynamic response sensitivity solution sequence. The procedure has the capability to solve optimization problems with frequency and static constraints in addition to forced response (displacement only) constraints. Implementation of the procedure resulted in a computer code, designated DYNOPT, for optimization of airframe structures under dynamic constraints.

The analysis block of the general methodology of Figure 1, on utilizing the DYNOPT program can subsequently be modified as shown in Figure 5. As can be noted, DYNOPT program can be modified by writing DMAP alter for dynamic stress constraints as well as a DMAP alter for associated sensitivity analysis. Furthermore, to bring in fatigue life into picture, a synthetic response needs to be constructed. Fatigue life will be computed for the MH-53J airframe components based on S-N curve methodology. Alternate fatigue life computation methods exist but would not be used for this work.
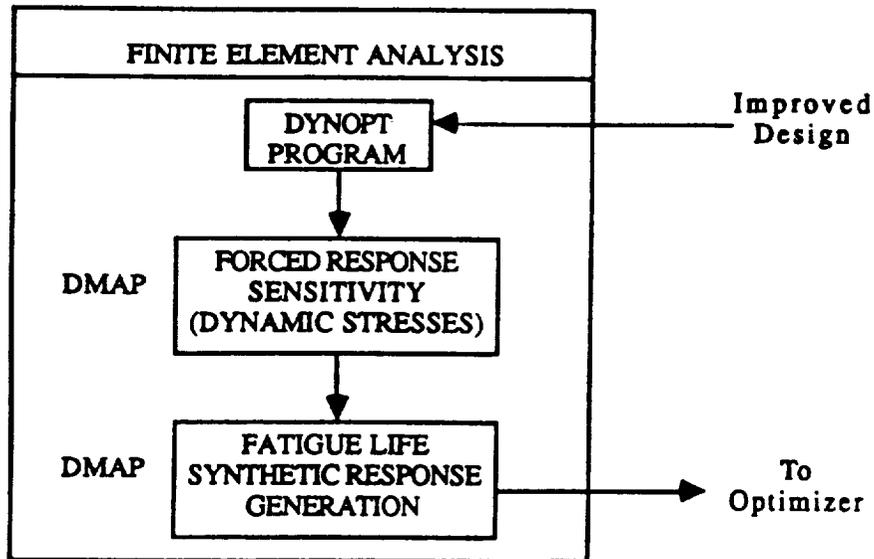
Figure 5. Modified Analysis Methodology

The selection of the regions considered for service life improvements are shown in Figure 6. These regions were determined from the 98 failure analysis reports for H-53 components and structure at Warner Robins Air Logistics Center (WRALC) to take advantage of the H-53 service history. The
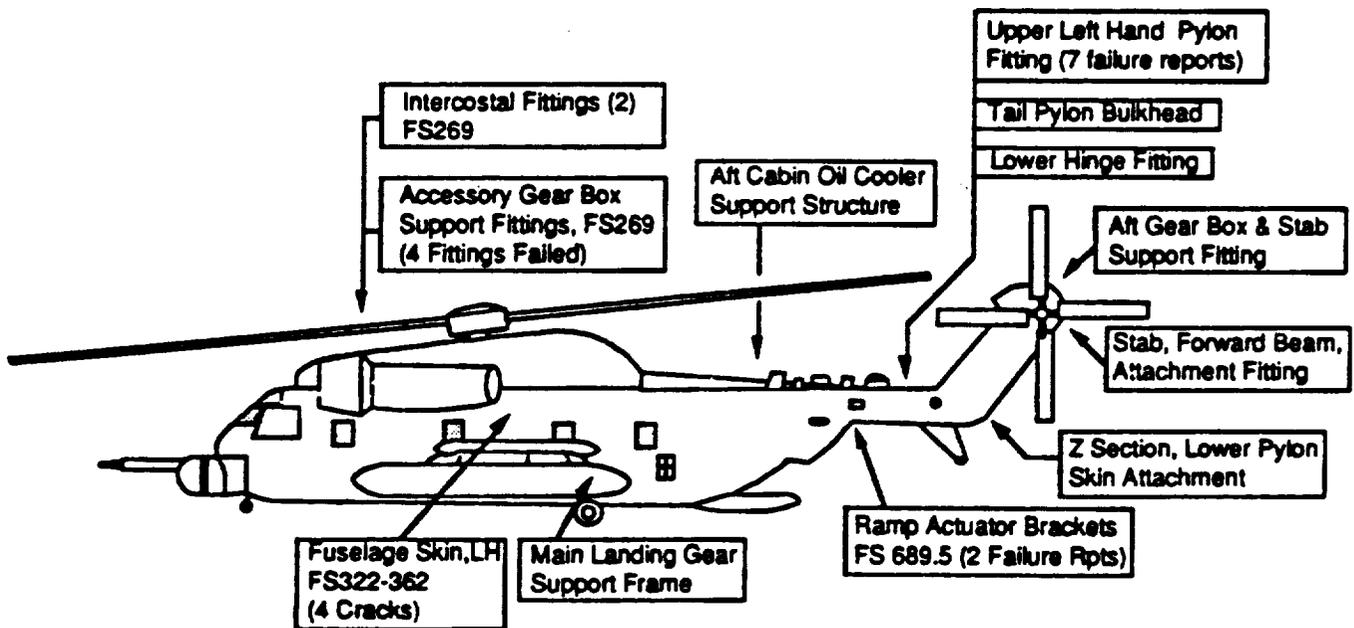
(Source: WRALC/MME)

Figure 6. Documented failures due to low stress, high cycle fatigue, airframe structure (Group 200).

percent of breakdown by fatigue phenomenon accounted for 40% of all the failures (Reference 7). The structure failures noted in the failure analysis reports were mapped onto the H-53 structure to determine the frequency and location of failure as an aid to locate potential "hot-spots" (Figure 6).

ANTICIPATED RESULTS

Modification of DYNOPT program is underway to develop DMAP alters for dynamic stress constraints and associated sensitivity analysis. Once the fatigue life synthetic response is formuated, the sensitivities of dynamic stresses at prescribed "hot-spots" can be computed with respect to the design variables of the design model. This sensitivity data will be used by the optimization algorithm along with the response information for dynamic structural modification of the MH-53J airframe. The final results will show the reduction in dynamic stresses with a corresponding increase in service life at the "hot-spots" after a certain number of iterations.

REFERENCES

1. Murthy, T.S.: Design Sensitivity Analysis of Rotorcraft Airframe Structures for Vibration Reduction, NASA/VPISU Symposium on Sensitivity Analysis in Engineering, Langley Research Center, Sept.25-26, 1986.

2. Adelman, H.M., and Mantay, W.R.: Integrated Multidisciplinary Optimization of Rotorcraft: A Plan for Development, NASA TM 101617 (AVSCOM TM 89-B-004), May 1989.

3. Kvaternik, R.G., Bartlett, F.D. Jr., and Cline, J.H.: A Summary of Recent NASA/Army Contributions to Rotorcraft Vibrations and Structural Dynamics Technology, Proceedings of the 1987 NASA/Army Rotorcraft Technology Conference, NASA CP 2495, pp 71-179.

4. Schmit, L.A., and Miura, H.: Approximation Concepts for Efficient Structural Synthesis, NASA CR-3226, December 1980.

5. MSC/NASTRAN Handbook for Structural Optimization (Preliminary), Version 66.

6. Murthy, T.S.: Application of DYNOPT Optimization Program for Tuning Frequencies of Helicopter Airframe Structures, Dynamics and Aeroelastic Stability Modeling of Rotorcraft Systems, Third Workshop, Army Research Office and Duke University, March 12-14, 1990.

7. Final Technical Report, GTRI Project A-8222, Specialized Engineering Services Supporting Development of a USAF MH-53J Aircraft Structural Integrity Program, WRALC contract No. F09603-85-G-3104-0034.

ACOUSTIC DESIGN CRITERIA IN A GENERAL SYSTEM FOR STRUCTURAL OPTIMIZATION

by

Torsten Bråmå
SAAB-SCANIA AB, Saab Aircraft Division
S - 581 88 LINKÖPING, SWEDEN

INTRODUCTION

Passenger comfort is of great importance in most transport vehicles. For instance, in the new generation of regional turboprop aircraft, a low noise level is vital to be competitive on the market. The possibilities to predict noise levels analytically has improved rapidly in recent years. This will make it possible to take acoustic design criteria into account in early project stages.

The development of the ASKA FE-system to include also acoustic analysis has been carried out at Saab Aircraft Division and the Aeronautical Research Institute of Sweden in a joint project. New finite elements have been developed to model the free fluid, porous damping materials and the interaction between the fluid and structural degrees of freedom. The FE approach to the acoustic analysis is best suited for lower frequencies up to a few hundred Hz. For accurate analysis of interior cabin noise, large 3-D FE-models are built but also 2-D models are considered to be useful for parametric studies and optimization.

The interest is here focused on the introduction of an acoustic design criteria in the general structural optimization system OPTSYS available at the Saab Aircraft Division. The first implementation addresses a somewhat limited class of problems. The problems solved are formulated; Minimize the structural weight by modifying the dimensions of the structure, while keeping the noise level in the cavity and other structural design criteria within specified limits.

THE OPTSYS SYSTEM

OPTSYS [1], developed together with the Aeronautical Research Institute, is a modular system with well defined interfaces to FE-programs (ASKA or ABAQUS) and codes for aeroelasticity. A mathematical programming approach is adopted were a sequence of convex approximations of the initial problem is solved, using the MMA method [2]. Gradients are calculated semi-analytically. Design variables associated to the shape of the structure, the element cross section properties or the material direction in the case of composite materials, can be treated. This approach makes it possible to take several different design criteria into account simultaneously. Constraints can so far be defined on displacement, stress, eigenfrequency, buckling, flutter and aileron efficiency. Other important ingredients are; the integration of a preprocessor to define shape variables, the treatment of discrete variables and the possibility to deal with substructured FE models. OPTSYS has been applied to both aerospace and automotive structures [3], e.g. an investigation of the potential weight savings in a composite wing of a fighter aircraft involving more than 700 design variables, simultaneous shape and thickness optimization of a Saab 9000 car suspension arm, detailed shape optimization of the cross section of a separation system for satellites to avoid stress concentration. OPTSYS originates from an early version of the OASIS system developed by Esping [4].

SENSITIVITY OF THE ACOUSTIC RESPONSE

The FE formulation leads to a symmetric matrix equation describing the coupled fluid–structure problem [5]. To avoid complex arithmetic in current prototype implementation, the excitation is restricted to the harmonic case and no porous elements are allowed in the model. If the damping effects are neglected the dynamic response problem can be written as ;

$$ (K - \omega^2 M) u = F $$

where ω is the excitation frequency, F is the load vector and u is the response vector. Vector u contains both the structural response and the acoustic pressure degrees of freedom. The equation can be solved directly in the primary degrees of freedom or by modal synthesis.

The derivative of the acoustic pressure p in a certain node with respect to a design variable x is calculated, using the adjoint method, according to;

$$ \frac{\partial p}{\partial x} = - \sum_e \left[u_e^t \left(\frac{\partial k}{\partial x} e - \omega^2 \frac{\partial m}{\partial x} e \right) v_e \right] $$

where k_e is the element stiffness matrix, m_e is the element mass matrix and v_e is the response vector corresponding to a unit load applied to the degree of freedom in which the acoustic constraint is defined. The summation is made over all elements associated to the design variable x.

The sound pressure level, SPL, is defined as;

$$ SPL = 20 \log \frac{|p|}{\sqrt{2} \, p_0} $$

where p is the peak value of the acoustic pressure in a specific location in the acoustic cavity and p_0 is the acoustic reference pressure provided by the user. The derivative of the sound pressure level is then given by;

$$ \frac{\partial (SPL)}{\partial x} = \frac{20 \log e}{p} \cdot \frac{\partial p}{\partial x} $$

The acoustic constraint, which is a special case of dynamic response, proved to be fairly easy to implement in OPTSYS. The treatment is similar to the case of displacement constraints. The character of the acoustic constraint is however not as attractive to deal with as the displacement constraint. The acoustic response is not a monotonous function of structural size variables, as a maximum will occur when a eigenfrequency gets close to the excitation frequency. This will lead to unconnected feasible regions in the design space which is a major difficulty for an optimization algorithm.

CAVITY WITH FLEXIBLE WALL

The cavity, FE model shown in figure 1, is surrounded by rigid walls on five sides. The rightmost wall, where the pressure is applied is however flexible. The excitation frequency is 60 Hz and the excitation force is applied as four point loads, F= 0.25 N , on the flexible wall corresponding to an external uniform pressure. Air density 1.205 kg/m^3 , speed of sound 340 m/s^2. Structural properties : E = $7.2 \, 10^{10}$ N $/m^2$, density = 2700 kg/m^3. The noise level inside the cavity was examined as a function of the thickness distribution in the flexible wall. The resulting maximum SPL in the cavity is illustrated in figures 2 and 4. Note the asymptotic behavior of the dynamic response when the eigenfrequency gets close to the excitation frequency.
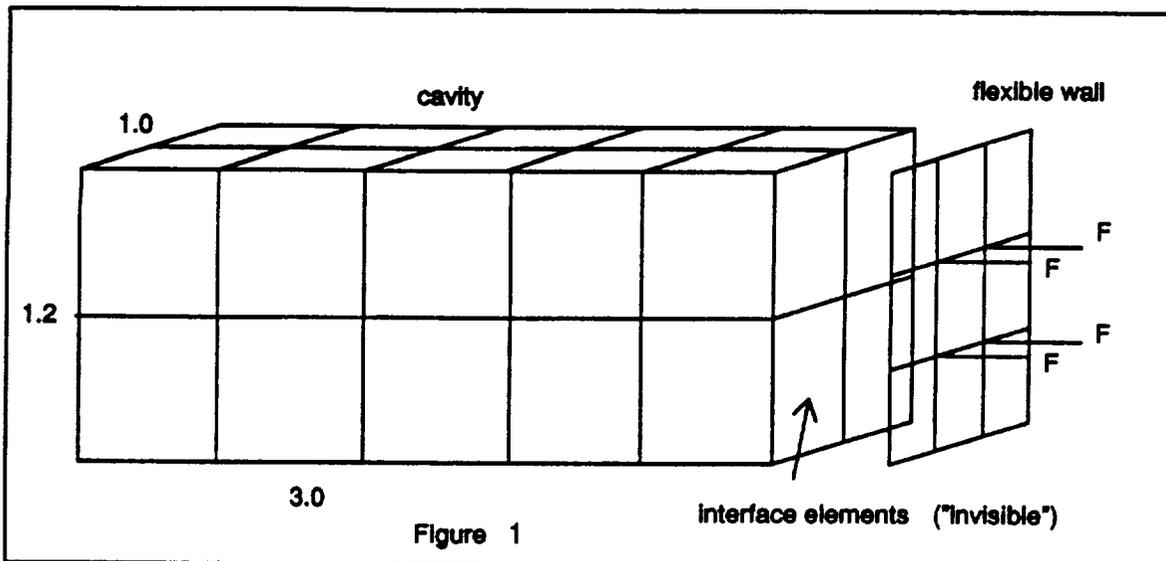
Figure 1

In the first optimization problem, see figure 2, one design variable was associated to the uniform thickness of the whole wall. Note that different solutions are found, depending on the starting point, due to the unconnected feasible regions.

As indicated in figure 3, a second design variable was associated to the center element. The iteration history in this two dimensional case is illustrated in figure 4. It can be noted that the constraint boundary corresponding to SPL = 110 dB seems to be very close to a straight line and that there is a small feasible region unconnected to the large one, which is not reached from the starting point. The final design is in this case bounded by the acoustic design criteria and the lower limit of the first design variable.

Figure 2

158

Linking of the elements
in the flexible wall to the
two design variables.

Figure 3

Figure 4
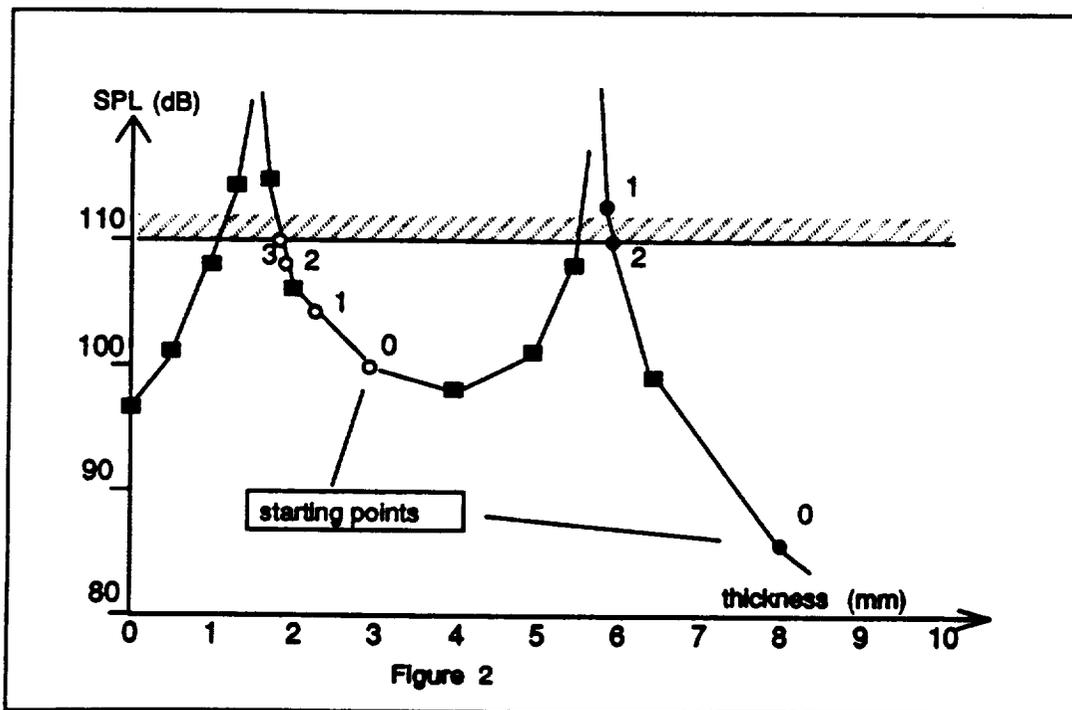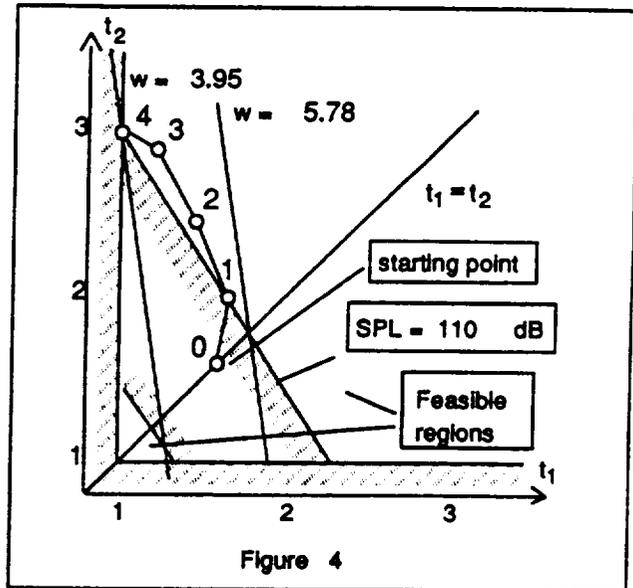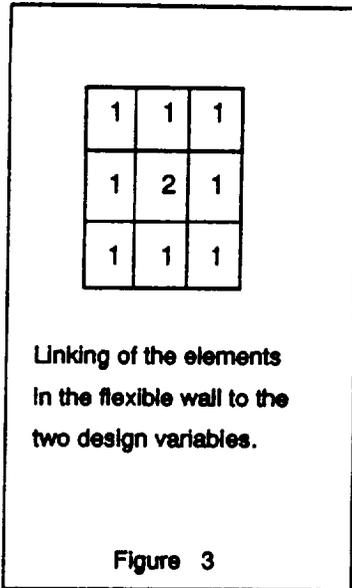
CROSS SECTION OF THE SAAB 340 AIRCRAFT

The 2–D FE–model , representing a cross section of the Saab 340 fuselage close to the plane of the propeller, consists of one substructure for the structural part and another substructure for the cavity, see figures 5 and 7. The cavity substructure contains 2–D acoustic elements and interface elements connecting the cavity model to the outer flange. The excitation force was simplified to be harmonic, i.e. the phase difference around the fuselage was neglected. The objective function is here the weight of the inner flange, i.e. the weight of the elements associated to design variables. The acoustic constraint is applied to three points in the cabin corresponding to measurement points in flight tests. The location of the design variables (1 – 37) and the constraint points are indicated in figure 6 .

FLA2

QUAD4

Corresponding FE model

Cross section of frame

Figure 5

Figure 6

159
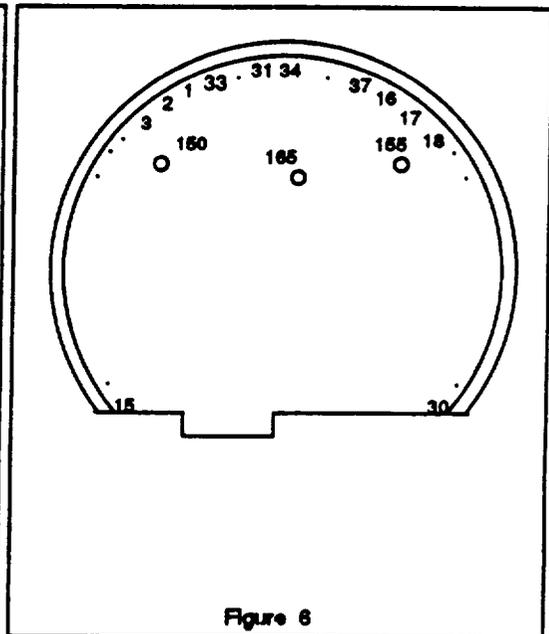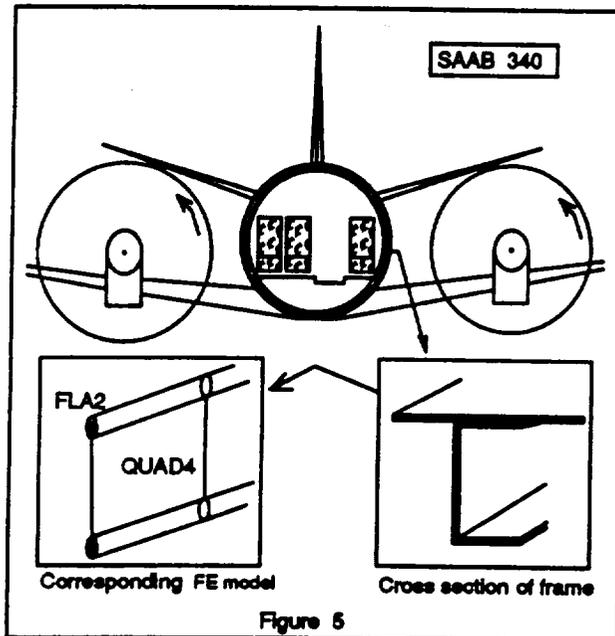
Two sequential optimization problems were solved. Figure 8 shows the iteration history for the objective and constraint functions. In phase I only increased flange areas were allowed. After six iterations the process converged without having reached the desired noise level as all variables giving a favorable contribution to the SPL have reached their upper limit. The additional 0.076 kg reduced however the noise level from 122.3 dB to 120.5 dB. In phase II the flange areas were also allowed to decrease. The desired noise level of 119 dB was reached after 4 iterations with a reduction in weight of 0.135 kg. The ASKA FE-model in figure 7 shows the final SPL distribution and the corresponding structural deformation for the final design. The picture looks very much the same for the initial design, the difference being that the noise level is lower for the final design. In the final design, material has been added in the region controlled by design variables 2 – 8 and removed elsewhere, making the structure less symmetric. The noise level in this test example is much higher than in the real aircraft and in a more realistic application of course other design criteria have to be considered as well.

Figure 7

160
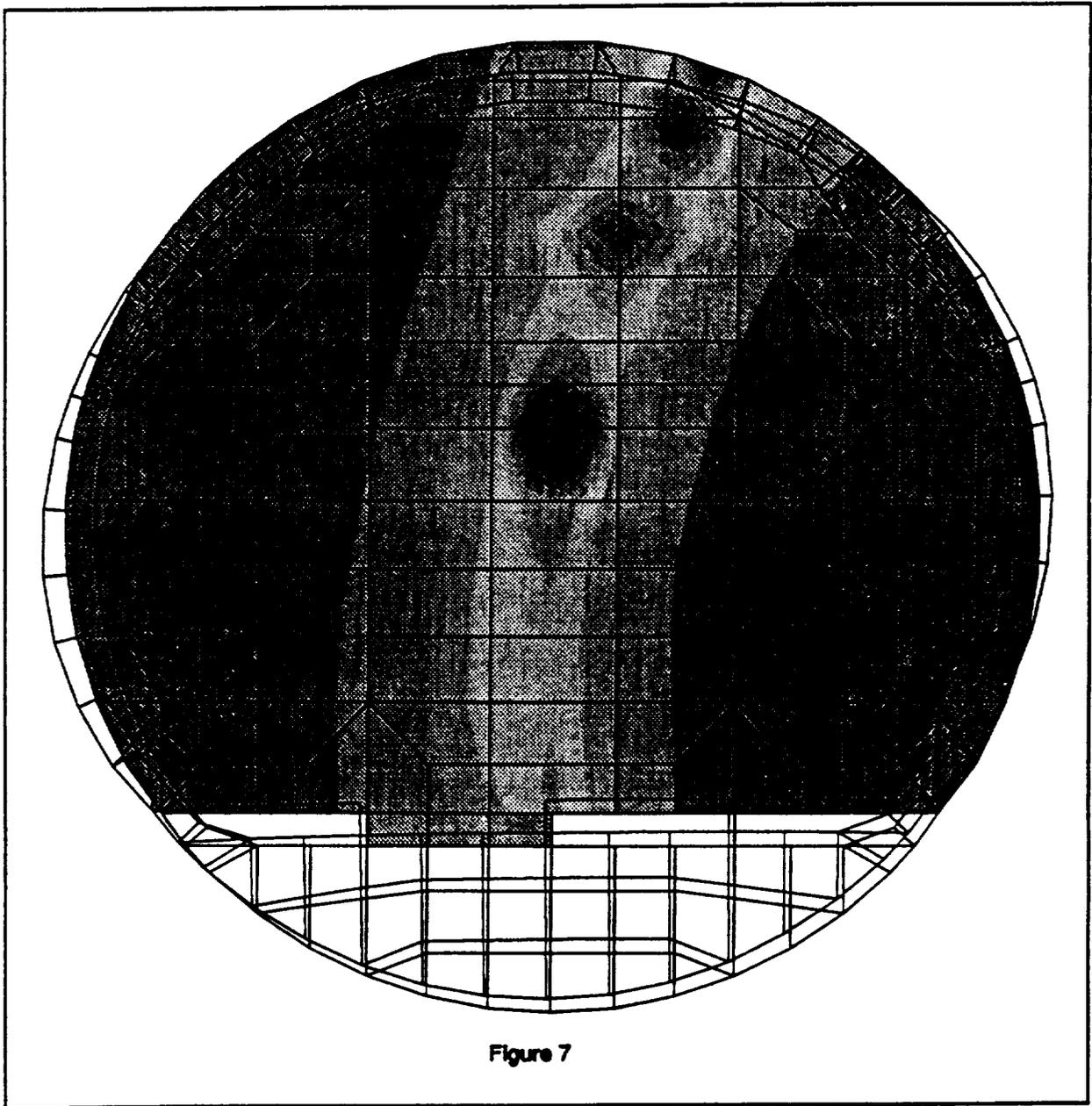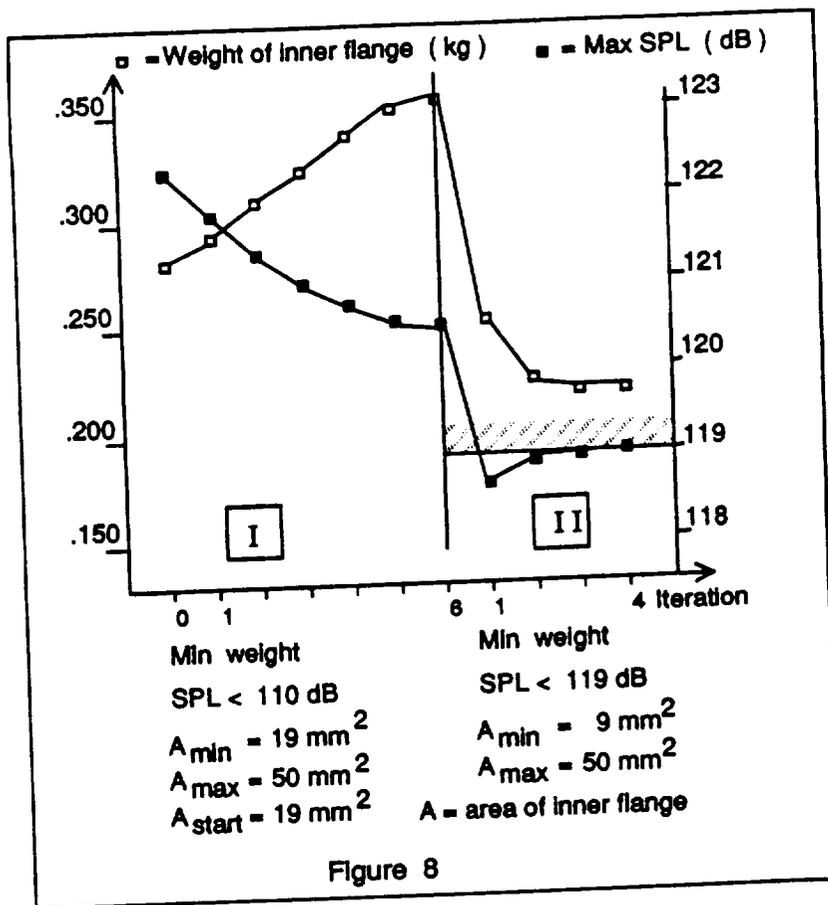
Figure 8

COMMENTS ON CURRENT AND FUTURE DEVELOPMENT

The current development includes complex excitation which is the case of the turboprop aircraft. It is also desirable to be able to deal with tuned dampers, heavy pieces of metal bolted to the fuselage with elastic damping elements, during optimization. A new finite element is being developed to model the damper and design variables can than be connected to its properties. The problem is to select the properties of the damper and to decide where to attach it to the structure. Another interesting future extension is for instance to associate design variables to the distribution of damping material.

REFERENCES

1) Bråmå, T. "Weight optimization of aircraft structures". Computer Aided Optimal Design: Structural and Mechanical Systems,(Ed. Mota Soares C.A.), pp. 971–985, Springer–Verlag, Berlin Heidelberg, 1987.

2) Svanberg, K. , "The Method of Moving Asymptotes – a new method for structural optimization", International journal for numerical methods in engineering, Vol. 24, pp. 359–373, 1987.

3) Bråmå, T. "Applications of the structural optimization program OPTSYS", ICAS–90–2.1.3 , 1990.

4) Esping, B.J.D. ,"The OASIS structural optimization system", Computers and Structures, vol. 23, No 3, pp. 365–377, 1986.

5) Sandberg, G. and Göransson, P. , "A Symmetric Finite Element Formulation for Acoustic Fluid–Structure Interaction Analysis", JSV, Vol. 123 (2), 1988.

OPTIMUM SEISMIC STRUCTURAL DESIGN
BASED ON RANDOM VIBRATION AND FUZZY GRADED DAMAGES

Franklin Y. Cheng* and Jin-ping Ou**

*Curators' Professor of Civil Engineering
University of Missouri-Rolla, Rolla, MO 65401 U.S.A.
**Research Professor of Engineering Mechanics
Harbin Architectural and Civil Engineering Institute
Harbin 150006, P.R. of China

ABSTRACT

This paper presents the fuzzy dynamical reliability and failure probability, as well as the basic principles and the analytical method of loss assessment for nonlinear seismic steel structures. Also presented is the optimization formulation and a numerical example for double objectives (initial construction cost and expected failure loss) and dynamical reliability constraints. The earthquake ground motion is based on a stationary filtered non-white noise and the fuzzy damage grade is described by damage index.

INTRODUCTION

A reasonable structural design should achieve both serviceability and economic objectives; optimum structural design is one of the means to achieve the objectives. Traditionally, structural optimum design is to minimize the initial construction cost under the constraints of structural safety. But, the optimum design of aseismic structures is not only to minimize the initial construction cost but also the failure loss of the structure damaged by earthquakes. For this purpose an optimization formulation of aseismic structures was presented in Ref. 3, based on the double objectives (initial construction cost and expected failure loss) and reliability constraints.

Since a structure under strong earthquakes will undergo inelastic deformation and dissipates hysteretic energy, earthquake ground motion is a random process, and the safety criterion of structure has some fuzziness, all these properties should be considered in the optimum design of aseismic structures and are therefore discussed in this paper.

STRUCTURAL MODEL AND MOTION EQUATION

For a SDOF steel structure shown in Fig. 1, the motion equation can be expressed as

$$m\ddot{X} + c_o\dot{X} + f_s(X,\dot{X}) = -mA(t) \tag{1}$$

in which the earthquake ground motion is simulated as a Gauss stationary filtered non-white noise with zero mean and power spectrum

$$S_A(\omega) = \frac{1+4\zeta_g^2 \frac{\omega^2}{\omega_g^2}}{4\zeta_g^2 \frac{\omega^2}{\omega_g^2} + (1-\frac{\omega^2}{\omega_g^2})^2} \quad \frac{S_o}{1+\frac{\omega^2}{\omega_h^2}} \tag{2}$$

where ζ_g and ω_g are respectively the damping ratio and predominant frequency of site soil, $\omega_h = 8\pi$ rad/s is a parameter of bedrock spectrum, and S_o is the spectral factor corresponding to seismic intensity degree. $f_s(X,\dot{X})$ is the restoring force of a structure and can be expressed by a bilinear model shown in Fig. 2, in which k is the primary elastic stiffness, α is the second stiffness coefficient and f_y is the yielding shear force. They are respectively obtained by the elastic and plastic analysis of structure as

$$k = \frac{24EI}{H^3} , \qquad f_y = \frac{4M_y}{H} \tag{3}$$

for the section shown in Fig. 3,

$$\alpha = \frac{\varphi_s - 1}{\frac{1}{\varphi_s^2} + \sqrt{\frac{h}{3}}(2 + \frac{1}{\varphi_s})\Big/(1 - \frac{1}{\varphi_s})\frac{\delta}{\varphi_s Z_y} - 1} \tag{4}$$

in which E = elastic modulus; I = moment of inertia of the cross section; and $\varphi_s = M_p/M_y$ is the shape factor, M_y $(\sigma_y Z_y)$ and M_p $(\sigma_y Z_p)$ are respectively the yielding and plastic moments; σ_y is the yielding stress; Z_y and Z_p are respectively the elastic and plastic section moduli. The symbols H, h and δ are given in Figs. 1 and 3.

RANDOM DEFORMATION AND HYSTERETIC ENERGY DISSIPATION

By means of the stochastic linearization technique, Eq. (1) can be written as following equivalent linear equation.

$$m\ddot{X} + c_e\dot{X} + k_eX = -mA(t) \tag{5}$$

in which c_e and k_e are respectively equivalent linear damping and stiffness, and $c_e = c_o + c_{ef}$. By using the energy balance technique of equivalent linearization and by assuming the displacement response $X(t)$ with random amplitude r, c_{ef} and k_e can be respectively obtained by (6)

$$c_{ef} = \int_0^\infty \bar{c}_{ef}(r)p(r)dr, \quad k_e = \int_0^\infty \bar{k}_e(r)p(r)dr \tag{6}$$

in which

$$\bar{c}_{ef}(r) = \begin{cases} \dfrac{4-(1-\alpha)(r-x_y)x_y}{\pi\omega_e r^2} k & (r > x_y) \\ 0 & (r \le x_y) \end{cases} \tag{7}$$

$$\bar{k}_e(r) = \begin{cases} k[\dfrac{1-\alpha}{\pi}\cos^{-1}(1-\dfrac{2x_y}{r}) + \alpha - \dfrac{2(1-\alpha)}{\pi r^2}(r-2x_y)\sqrt{x_y(r-x_y)}] & (r > x_y) \\ k & (r \le x_y) \end{cases} \tag{8}$$

$$p(r) = \frac{r}{\sigma_x^2}\exp(-\frac{r^2}{2\sigma_x^2}) \tag{9}$$

where x_y is the yielding displacement of structure, $\omega_e = \sigma_{\dot{x}}/\sigma_x$, σ_x^2 and $\sigma_{\dot{x}}^2$ are the stationary variances of the displacement and velocity responses. By introducing the state vector $\bar{Z} = [X, u, \dot{X}, \dot{u}, \omega]$, the covariance matrix $\Gamma = [Z(t)Z^T(t)]$ can be obtained by the following algebraic matrix equation

163

$$\bar{A}\bar{\Gamma} + \bar{\Gamma}\bar{A}^T = \bar{D} \tag{10}$$

in which

$$\bar{A} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ \dfrac{k_e}{m} & -\omega_g^2 & \dfrac{c_e}{m} & -2\zeta_g\omega_g & 0 \\ 0 & \omega_g^2 & 0 & 2\zeta_g\omega_g & \omega_h \\ 0 & 0 & 0 & 0 & \omega_h \end{bmatrix}, \quad \bar{D} = \begin{bmatrix} \bar{0}_4 & \bar{0}_4 \\ \bar{0}_4^T & 2\pi S_o \end{bmatrix}$$

where $\bar{0}_4$ and $\bar{0}_4$ are respectively 4th order matrix and vector.

The correlation function matrix $\bar{R}(\tau)$ can be obtained by

$$\bar{R}(\tau) = \bar{\Gamma}\bar{\Phi}^T(\tau) \tag{11}$$

in which $\bar{\Phi}^T(\tau)$ is the transfer matrix of state vector $Z(t)$. Let $X_m = \max_{0 \le t \le T} X(t)$ be the structure's maximum displacement, in which T is the duration of earthquake, the mean and variance of X_m can be respectively obtained by

$$E[X_m] = \left(\sqrt{2\ell_n(\nu_o T)} + \frac{0.5772}{\sqrt{2\ell_n(\nu_o T)}} \right)\sigma_x \tag{12}$$

$$\sigma_{X_m}^2 = \frac{\pi^2}{12\ell_n(\nu_o T)}\sigma_x^2, \quad \nu_o = \frac{\omega_e}{\pi}. \tag{13}$$

Let $\varepsilon(t)$ be the hysteretic energy dissipated of a structure during the earthquake, its mean and variance can be respectively obtained by

$$E[\varepsilon(T)] = c_{ef}\sigma_{\dot{x}}^2 T \tag{14}$$

$$V[\varepsilon(T)] = 4c_{ef}^2 \int_o^T (T-\tau)R_{\dot{x}}^2(\tau)d\tau \tag{15}$$

in which $R_{\dot{x}}^2(\tau)$ is the correlation function of the velocity response $\dot{X}(t)$.

DYNAMICAL RELIABILITY BASED ON FUZZY GRADED DAMAGES

Damage Index - Taking the influence of both the deformation and the hysteretic energy dissipated on the damage of a structure, a damage index of the hysteretic steel structure can be defined as

$$D_c = \left(\frac{X_m}{x_u}\right)^\beta + \left(\frac{\varepsilon(T)}{\varepsilon_u}\right)^\beta \tag{16}$$

in which X_m and $\varepsilon(T)$ are respectively the maximum displacement and hysteretic energy dissipated of a structure during the earthquake, x_u and ε_u are respectively the ultimate displacement and ultimate hysteretic energy dissipated of a structure, $\beta \ge 1$, a factor determined by experiments.

164

Using plastic analysis and considering the buckling of column, we can obtain the ultimate displacement as follows:

$$x_u = \begin{cases} f_u/k & (f_u \leq f_y) \\ x_y + (f_u - f_y)/\alpha k & (f_u > f_y) \end{cases} \tag{17}$$

in which $f_u = 4M_u/H$, M_u is the ultimate moment which is given in Eq. 18 with the axial force effect,

$$M_u = \begin{cases} M_p & (0 \leq N \leq 0.15N_y) \\ 1.18(1 - \dfrac{N}{N_y})M_p & (0.15N_y < N \leq N_y) \end{cases} \tag{18}$$

where N is the axial load, $N_y = \sigma_y A$, A is the cross-sectional area.

By using the low cycle fatigue experimental results of steel structures (5), the ultimate hysteretic energy dissipation can be expressed as

$$\varepsilon_u = 2(1-\alpha)E[\mu]f_y x_y (2N_f) \tag{19}$$

in which $E[\mu]$ is the plastic ductility and N_f is the number of full cycles, they are respectively obtained by

$$E[\mu] = \frac{E[X_m] - x_y}{x_y} \tag{20}$$

$$2N_f = (\frac{E[\mu]}{\mu_u})^{- 1/0.6} \tag{21}$$

where μ_u is the ultimate plastic ductility, i.e.

$$\mu_u = \frac{x_u - x_y}{x_y} \tag{22}$$

Fuzzy Damage Grades - The damage of a structure under earthquake is often divided into the following five fuzzy grades:

$$[\underset{\sim}{B}_1, \underset{\sim}{B}_2, \underset{\sim}{B}_3, \underset{\sim}{B}_4, \underset{\sim}{B}_5] = [\text{Slight, Minor, Moderate, Severe, Collapsed}] \tag{23}$$

Obviously, all these grades possess strong fuzziness in their definition. If the damage index D_c is taken to express above grades, then $\underset{\sim}{B}_i$ (i = 1, 2, ..., 5) should be a fuzzy subset on the value region of D_c and has the membership function as shown in Fig. 4.

Let $\underset{\sim}{B}_i^*$ represent the fuzzy state region in which $\underset{\sim}{B}_i$ or more severe damage will not occur for the structure, its membership curve should have the form as shown in Fig. 5 and may be expressed by

$$\mu_{\underset{\sim}{B}_i^*}(d) = \begin{cases} 1 & (d \leq d_{i-1}) \\ \dfrac{1}{2}[1 - \sin(\dfrac{d-d_{i-1}}{d_i-d_{i-1}} - \dfrac{1}{2})\pi] & (d_{i-1} < d \leq d_i) \\ 0 & (d > d_i) \end{cases} \tag{24}$$

in which d_i (i = 1, 2, ..., 5) are the parameters of fuzzy damage grades and their values are shown in Table 1.(2)

165

<u>Dynamical Reliability and Failure Probability</u> - Based on the fuzzy graded damage and fuzzy safe region shown in Table 1, the dynamical reliability that the structure will not suffer B_{-i} or more severe damage can be obtained by

$$P_s(B^*_{-i}) = \int_o^{d_i} f_{D_c}(d)\mu_{B^*_{-i}}(d)dd \tag{25}$$

in which $f_{D_c}(d)$ is the probability density function of the damage index D_c and assumed as the extreme value distribution I, i.e.

$$f_{D_c}(d) = \frac{1}{\alpha}\exp[-\frac{d-\beta}{\alpha}] \exp[-\exp(-\frac{d-\beta}{\alpha})] \tag{26}$$

in which

$$\alpha = \sigma_{D_c}/1.2826, \quad \beta = \langle D_c\rangle - 0.5772\alpha \tag{27}$$

where $\langle D_c\rangle$ and $\sigma^2_{D_c}$ are respectively the mean and variance of D_c and approximately obtained as

$$\langle D_c\rangle = (\frac{E[X_m]}{x_u})\beta + (\frac{E[\epsilon(T)]}{\epsilon_u})\beta \tag{28}$$

$$\sigma^2_{D_c} = (\frac{\sigma^2_{X_m}}{x_u^2})\beta + (\frac{\sigma^2_{\epsilon(T)}}{\epsilon_u^2})\beta \tag{29}$$

Obviously, the failure probability that the structure will suffer B_{-i} or more severe damage is

$$P_f(\bar{B}^*_{-i}) = 1 - P_s(B^*_{-i}) \tag{30}$$

Then, the failure probability the structure will suffer only B_{-i} graded damage is

$$P_f(B^*_{-i}) = P_f(\bar{B}^*_{-i}) - (\bar{B}^*_{-i+1}) \tag{31}$$

in which i = 1, 2, ..., 5, and $P_f(\bar{B}^*_{-6}) = 0$

LOSS ASSESSMENT

<u>Basic Principles of Loss Assessment</u> - According to the definition of damage grades, we can put forward some basic principles to the loss assessment of structures.

The loss to the slight or minor damage includes only structural repair cost.

The loss to the moderate damage includes structural repair and replacement cost.

The loss to the severe damage includes additional replacement cost; the loss also includes property damage but excludes the loss of expensive equipments. However, the liability lcss due to minor or serious injuries of the people and loss due to business interruption are included.

The loss to the collapsed damage includes additional replacement cost, the loss of all equipment, liabilities due to death or serious injuries of the people and loss due to business interruption.

According to the economic condition in China, all losses mentioned above are estimated based on the following assumption: The losses of structural repair corresponding to the slight and minor damage are respectively assumed to be 15% and 35% of the initial construction cost. The loss of structural repair and replacement corresponding to the moderate damage is assumed as 85% of the initial construction cost. The additional replacement cost is about twice the initial construction cost. The loss due to property damage, excluding the loss of expensive equipment, is taken as 25% of all equipment cost. Loss due to death is calculated based on an average death age of 30 and is the sum of the person's salary until he reaches the retirement age of 60 years. Thus, this loss is 30 times the average annual net income (approximately 2,000 Chinese Yuns). The loss due to serious injury is assumed to be 30,000 Yuns per person. The loss due to minor injury is 2,000 Yuns per person. Business interruption is estimated as the net income of structural service during a reconstruction period. The loss due to legal service may be assumed to be 15% of the total failure loss.

Based on the principles mentioned above, let $L_f(\underline{B}_i)$ be the loss corresponding to the damage grade $\underline{B}_i$ ($i = 1, 2, \ldots, 5$), we have

$$L_f(\underline{B}_1) = 0.15\ C_I, \quad L_f(\underline{B}_2) = 0.35\ C_I, \quad L_f(\underline{B}_3) = 0.85\ C_I$$

$$L_f(\underline{B}_4) = 1.15\ (2C_I + 0.25C_e + 2000N_m + 30000\ N_{s4} + R_p C_s)$$

$$L_f(\underline{B}_5) = 1.15\ (2C_I + C_e + 30000\ N_{s5} + 60000N_d + R_p C_s)$$

$$(32)$$

in which C_I is the initial construction cost, C_e is the cost of all equipment, C_s is the net income of structural service per year, R_p is the reconstruction period (years) of the structure, N_m and N_d are respectively the number of minor injury and death of people, N_{s4} and N_{s5} are the number of serious injury people respectively corresponding to the damage grades $\underline{B}_4$ and $\underline{B}_5$. According to the statistical results of the injury and death of people due to earthquake in China, we approximately take

$$N_m = 0.05\ N_p, \quad N_d = 0.15\ N_p$$

$$N_{s4} = 0.02\ N_p, \quad N_{s5} = 0.35\ N_p$$

$$(33)$$

in which N_p is the mean number of the people in the building.

<u>Expected failure loss</u> - By using the results of failure probability $P_f(\underline{B}_i)$ and estimated loss $L_f(\underline{B}_i)$ ($i = 1, 2, \ldots, 5$), we can obtain the expected failure loss of the structure

$$C_f = \sum_{i=1}^{5} L_f(\underline{B}_i) P_f(\underline{B}_i)$$

$$(34)$$

OPTIMIZATION FORMULATION

By taking the sum of initial construction cost and expected failure loss as the objective function of structural design and considering the reliability constraints of structure, the optimum design of structure can be formulated as (3)

$$\left.\begin{array}{ll} \text{Find} & \text{I} \\[2mm] \text{Min} & C_T = C_I + C_f \\[2mm] \text{S.t.} & P_s(\underline{B}_{-i}^*) \geq P_{si}^* \end{array}\right\} \tag{35}$$

in which I is the cross sectional moment of inertia, C_I is the initial construction cost, C_f is the expected failure loss, $P_s(\underline{B}_{-i}^*)$ is the reliability that the structure will not suffer $\underline{B}_{-i}$ or more severe damage, P_{si}^* is the lowest allowable reliability, and the index i in the constraint equation that signifies the relationship with the allowable damage extent of the structure under earthquake.

NUMERICAL EXAMPLE

The SDOF steel structure shown in Fig. 1 is designed based on the following seismic input: earthquake with intensity degree of 8, $S_o = 63.98$ cm^2/s^3, T = 7.0 s, $\zeta_g = 0.72$, and $\omega_g = 20.94$ rad/s. Using structural properties given in Table 2, and the column's cross-section shown in Fig. 3, the cross sectional area, elastic and plastic section moduli have the following relationships:

$$A = 0.8I^{0.5}, \quad z_y = 0.78 \ I^{0.75}, \quad z_p = I^{0.75}$$

For the structural losses assumed in Table 3, and the initial construction cost is taken as

$$C_I = 1000 \ C_g$$

in which C_g is the cost of structural steel, then, based on the optimization formulation in Eq. 35 and by taking $\beta = 1.0$, i = 4, $P_{s4}^* = 0.9950$, we can obtain the relationship curve between the objective function C_T and the cross sectional moment of inertia I, which is shown in Fig. 6. The numerical results are given in Table 4. The optimum design results are I* = 8400 cm^4 and $C_T^* = 428300$ Yuns.

CONCLUSION

This paper presents the fuzzy damage grades described by the damage index. It also presents the basic principles for the structural loss assessment corresponding to the grades based on the economical condition in China, for which the loss functions to the damage grades are given. An optimization formulation and a numerical example of the nonlinear aseismic steel structure are shown on the basis of double objectives (initial construction cost and expected failure loss) and dynamical reliability constraints.

REFERENCES

(1) Ou, Jinping and Niu, Ditao, "Parameters in the Random Process Models of Earthquake Ground Motion and their Effects on the Response of Structures," Journal of Harbin Architectural and Civil Engineering Institute, Vol. 23, No. 2, 1990, (in Chinese).

(2) Ou, Jinping, Niu, Ditao, and Cheng, Franklin Y., "Comprehensive Damage Index of Nonlinear Steel Structures Subjected to Earthquake", Journal of Building Structures, to be published, (in Chinese).

(3) Cheng, Franklin Y., and Chang, Chein-Chi, "Optimum Studies of Coefficient Variation of UBC Seismic Forces," <u>Structural Safety & Reliability</u>, Vol. III, Proceedings of ICOSSAR '89, pp. 1895-1902.

(4) Ou, Jinping, "Fuzzy Random Vibration," Doctorate Dissertation of Harbin Architectural and Civil Engineering Institute, 1987, (in Chinese).

(5) McCabe, S.L., and Hall, W.J., "Assessment of Seismic Structural Damage", <u>Journal of Structural Engineering</u>, ASCE, Vol. 115, No. 9, 1989.

(6) Satsuya Soda and Sukenobu, Tani, "Dynamic Analysis of Elastic-Plastic Structures by Statistical Equivalent Linearization Method," <u>Proc. of Architectural Society of Japan</u>, No. 283, 1979, (in Japanese).

TABLE 1.　PARAMETERS OF FUZZY DAMAGE GRADES

d_1	d_2	d_3	d_4	d_5
0.10	0.25	0.45	0.65	0.90

TABLE 2.　PARAMETERS OF STRUCTURE

m (kg s^2/cm)	C_o (kg s/cm)	H (cm)	E (kg/cm^2)	σ_y (kg/cm^2)
65	100	400	2×10^6	2400

TABLE 3.　PARAMETERS OF LOSS ASSESSMENT

C_e (million Yun)	C_s (million Yun)	R_c (year)	N_m (person)	N_{s4} (person)	N_{s5} (person)	N_s (person)
0.10	0.20	1	50	20	350	150

TABLE 4.　RELATIONSHIP BETWEEN THE SECTION AND THE COST

Section I / Cost	8000 (cm^4)	8200 (cm^4)	8400 (cm^4)	8600 (cm^4)	8800 (cm^4)	9000 (cm^4)
C_I (million Yun)	0.2824	0.2859	0.2894	0.2928	0.2962	0.2995
C_f (million Yun)	0.1465	0.1426	0.1389	0.1358	0.1328	0.1302
C_T (million Yun)	0.4289	0.4285	0.4283	0.4286	0.4290	0.4297
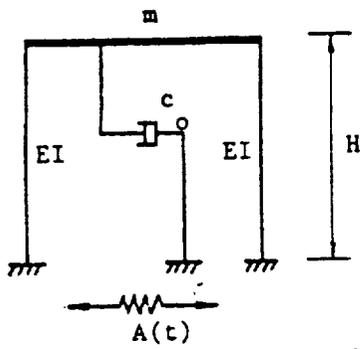
Fig. 1. Structural model   Fig. 2. Restoring force model   Fig. 3. Cross section
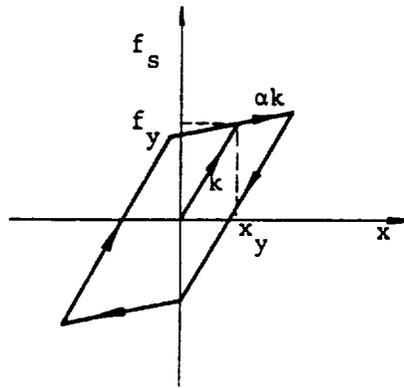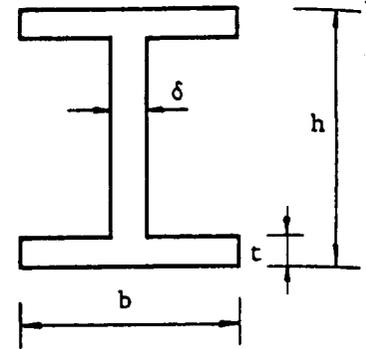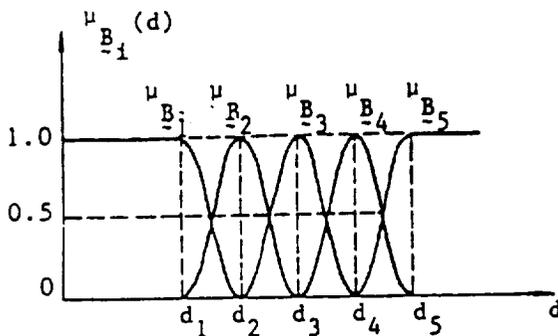
Fig. 4. Membership function of fuzzy
         damage grades
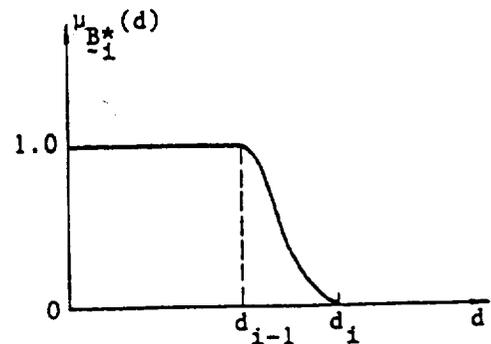
Fig. 5. Membership function
         of fuzzy safe region
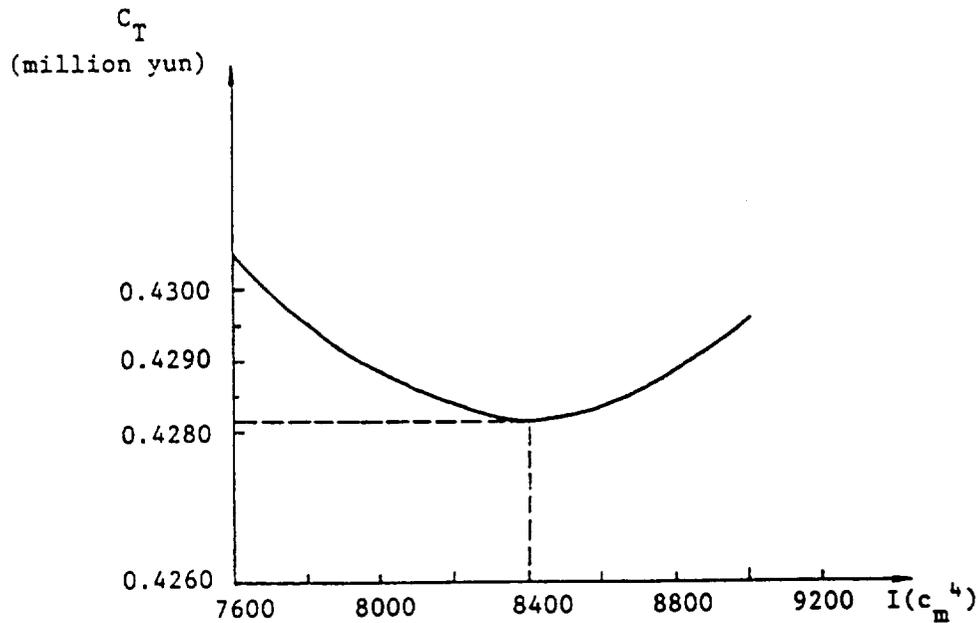
Fig. 6. Cross section (I) vs. cost (C_T)

APPLICATION AND ASSESSMENT OF STRUCTURAL OPTIMIZATION AND ACTIVE CONTROL FOR SEISMIC STRUCTURES

by

N94-71439

FRANKLIN Y. CHENG
Curators' Professor of Civil Engineering
University of Missouri-Rolla
Rolla, MO   65401   U.S.A.

ABSTRACT

This paper first presents a brief review on the application of optimization and active control of seismic structures along with some of the author's recent work. It then assesses the practicality and future development of seismic structural optimization, and some practical problems associated with active control.

REVIEW OF SOME APPLICATIONS

Since the author has reviewed the subject in 1983 (6) and 1985 (12), this article is intended to highlight some recent results and includes only pertinent publications after the year 1980. Generally, the literature review can be divided into the following categories: 1) nature of structures, such as frameworks, buildings, and bridges; 2) construction materials, such as steel, reinforced concrete, and mixed steel and concrete; 3) objective functions of minimum weight or cost; 4) optimization methods; 5) deterministic or nondeterministic in response and resistance; and 6) optimum control. Apparently, the above categories are quite nebulous and they overlap, and optimum control could be in an independent category.

Most structures that are optimized are frameworks (1, 2, 4, 5, 11, 13), only a few are building systems (8, 10, 38), and bridges (42). Major research work is for steel structures, a few are for reinforced concrete (1, 2, 21, 26), and very few are for mixed steel and reinforced concrete construction (3, 45). References 9, 15, 25 are based on minimum cost; the others are based on minimum weight or other types of objective functions. Most of the results were obtained by using conventional mathematical programming. A few were based on analytical procedures and iteration schemes. However, References 4, 7, 14, 16, 22, 40 were based on the modern optimization techniques currently in vogue. Other notable works related to optimization are: based-isolation (28, 30, 31), friction-joint (36), mode- and active-control (19, 20, 34, 35, 41, 43, 44), and fuzzy applications. (27, 29)

This presentation is summarized in the following classifications as 1) deterministic structures, 2) nondeterministic structures, 3) active control, and 4) structural optimization with active control.

Deterministic structures - A number of publications are dealing with deterministic systems for mathematical models or parameter investigations, or both. Refs. 9, 13, 38 are not only the mathematical developments, but also have associated computer programs for both parameter studies and practical structural design of 2-D and 3-D structural systems. The 2-D structures are formulated on the basis of the displacement method and the consistent mass model with the second-order P-Δ effect. The structural systems are trusses, and unbraced and braced frames of single-, double-, k-, and eccentric-bracings. The dynamic

forces may be 1) seismic excitations at the base with either one horizontal or one horizontal and one vertical earthquake motion, and 2) dynamic forces or wind forces at the structural nodes. The earthquake motions include 1) the actual earthquake records, 2) various response spectra, 3) building code provisions in the U.S. and abroad with and without soil-structure interaction. The structural members are either built-up sections or hot-rolled wide flange sections. The objective functions can be the minimum weight or minimum cost of a structural system. For 3-D structures, the structural elements are steel and reinforced concrete members. Seismic input includes response spectra and code provisions. The response spectra are developed for multicomponent excitations. The provisions include the equivalent lateral force technique and the modal analysis procedure. For both 2-D and 3-D structural optimization, various static loading conditions are included.

Nondeterministic structures - Several references (7, 15, 18, 26, 33, 41, 42) are related to nondeterministic structures. UMR's work was emphasized on the optimality criteria developments, formulation of structural response and resistance, and parameter studies (7, 15, 24). Three types of seismic loading models were employed in the study: UBC codified forces, Newmark's nondeterministic seismic response spectra for both horizontal and vertical ground motions, and Gaussian random process with a constant or varied seismic spectra. Four live loading models were also used in the study: ANSI (American National Standard Institute), NBS (National Bureau of Standards), UK (United Kingdom), and UNREDUCED (actual) models.

The cost objective function has initial construction cost (C_I), future failure cost (L_f), and system probability of failure (P_{fT}). The total cost is expressed as

$$C_T = C_I + L_f \, P_{fT} \qquad (1)$$

for which the reliability is based on normal and lognormal distribution with two different 1st and 2nd variance approaches.

Active control - Notable review on active control has been reported by Soong (39) from which one may find various theoretical and experimental studies of the control system. UMR's work has dealt with the issue of whether the placement of actuators at certain location of a structure more advantageous than for those at other location (37). The term optimal actuators placement reflects upon the reduction of the structure's response while using the minimum control effort. The UMR studies include the minimization of a control energy performance index; minimization of response performance index; and maximizing a controllability performance index. The three methods are compared and simulation studies are carried out using various earthquake records. The results are useful and practical.

UMR's studies also dealt with another issue; that is in application of structural control, it may be necessary to limit not only the magnitude of the control force, but also the control force time-rate at which we demand the control force to be supplied (20, 25). This is especially important when large magnitudes of control forces are required. Two optimal control algorithms were developed. The first algorithm is an extension of the traditional quadratic performance index to include a control derivative term. The second algorithm, in addition to introducing a control derivative term in the quadratic performance index, imposes a constraint on the upper bound of the control rate. The results show that the control rate can be prespecified and practically necessary.

172

Structural optimization with active control - Structural optimization allows one to design and construct a structure to satisfy an economical and serviceable objective. The application of active control is used as additional redundancy to minimize the deformation of structure for safety and serviceability; the structure itself, however, is designed based on conventional trial and error procedures but not the optimum. It is therefore UMR's work has combined structural optimization with control (20, 23). The advantages of the combination approach are that it can have all the strong points of both structural optimization and optimal control. The studies dealt with the optimal design of building structures equipped with active control systems of mass damper, tendon, and a combination of both. The structural optimization is formulated as a constrained minimization problem for which the design variables are the floor stiffnesses of the building and certain control parameters. The constraints include floor drifts, floor displacements, control forces, and natural frequencies. A control energy performance index is also minimized to find optimal weighting matrices that yield the least optimal control forces satisfying the constraints.

CONCLUDING REMARKS ON ASSESSMENT OF PRACTICALITY AND PROBLEMS

Structural optimization is a scientific approach which is remarkably different from the conventional design that is based on trial and error procedures. Structural optimization has been in practical use in many engineering disciplines. For seismic structures, a great deal of research results and a few large capacity computer programs are available for practical use in engineering practice. However, much more research work is still needed in developing more efficient algorithms for multiple objective functions and multiple damage levels associated with low-, median-, and severe earthquakes for both deterministic and nondeterministic systems.

Although the application of active control in civil engineering has been investigated in the past two decades and the recent research results show that the system is very promising in preventing structural damages, the practical use of the control in real engineering construction has some problems which are worthwhile to be noted. They are 1) back-up energy supplies to ensure the control system remains to be operational during the earthquakes; 2) discrepancy between the mathematical structural model and the actual structures such as how to include structural members' inelastic behavior or failure into the feedback system; 3) the effect of multicomponent seismic input on structural response because the earthquake motion has six components of three linear, one torsional, and two rocking motions; and 4) the effect of soil-structure interaction. Some of the aforementioned problems are currently being studied by various researchers, including the author. A great deal of research efforts must be carried through national and international cooperation in algorithm developments, laboratory tests, and full scale verifications before the system can be practically implemented in engineering practice.

ACKNOWLEDGMENTS

REFERENCES

1. Austin, A.M. and Pister, K.S., "Optimization-Based Computer-Aided Design of Earthquake Resistant Steel Structures", 8th WCEE Proceedings, Vol. 5, 451-458, 1984.

2. Balling. R.J., Pister, K.S., and Ciampi, V., "Optimal Seismic-Resistant Design of a Planar Steel Frame", _Earthquake Engineering and Structural Dynamics_, Vol. 11, 541-556, 1983.

3. Braga, F., D'Asdia, P., Dolce, M., "Optimum Design of Aseismic Coupled Shear Walls", _7th WCEE Proceedings_, Vol. 4, 493-496, 1980.

4. Cheng, F.Y., "Energy Distribution Criteria for Braced and Unbraced Structural Design Subjected to Parametric Earthquake Motions," _Proceedings of the 7th World Conference on Earthquake Engineering_, Vol. 6, pp. 665-672, 1980.

5. Cheng, F.Y., "Evaluation of Frame Systems Based on Optimality Criteria with Multicomponent Seismic Inputs, Performance Constraints, and P-Δ Effect", _Optimization of Distributed Parameter Structural Systems_, Sijthoff and Noordhoff International, Vol. 1, 650-684, 1981.

6. Cheng, F.Y., "Design Algorithms for Static and Dynamic Structures Based on Optimality Criteria," _Proceedings of the U.S.-China Workshop on Advances in Computational Engineering Mechanics_, 1983.

7. Cheng, F.Y., "Optimum Design of Seismic Structures With Risk Consideration", Paper No. 10, _U.S.-Japan Cooperative Research Proceedings_, 1983.

8. Cheng, F.Y. and Truman, K.Z., "Optimization Algorithm of 3-D Building Systems for Static and Seismic Loadings", _Modeling and Simulation in Engineering_, North-Holland Publishing, 315-326, 1983.

9. Cheng, F.Y., "Optimum Design of 2-D and 3-D Structures Subject to Multicomponent Seismic Excitations," _Proceedings of the AIT-CCNAA (U.S.-Taiwan) Joint Seminar on Research for Multiple Hazards Mitigation_, pp. 182-202, 1984.

10. Cheng, F.Y. and Truman, K.Z., "Optimum Design of Steel and Reinforced Concrete 3-D Seismic Building Systems," _8th WCEE Proceedings_, Vol. 5, pp. 475-482, 1984.

11. Cheng, F.Y. and Juang, D.S., "Assessment of ATC-03 for Steel Structures Based on Optimization Algorithm," _8th WCEE Proceedings_, Vol. 5, pp. 435-442, 1984.

12. Cheng, F.Y., "Developments in Optimal Seismic Structural Design," _Proceedings of U.S.-China-Japan Trilateral Symposium/Workshop on Engineering for Multiple Natural Hazard Mitigation_, pp. E10-1-24, 1985.

13. Cheng, F.Y. and Juang, D.S., _ODSEWS-2D-II - User's Manual for Optimum Design of 2-Dimensional Steel Structures for Static, Earthquake, and Wind Forces--Version II_, NSF Report, the U.S. Commerce Department, Virginia, NTIS PB87-163093/AS (134 pages), 1985.

14. Cheng, F.Y., and Juang, D.S., "Computer Aided Structural Optimization for Static and Wind Forces," _Proc. of 5th U.S. National Conf. on Wind Eng._, 1985.

15. Cheng, F.Y. and Chang, C.C., "Optimum Design of Steel Building with Consideration of Reliability," _Proceedings of the 4th International Symposium on Structural Safety and Reliability_, Vol. IV, pp. 81-90, 1985.

16. Cheng, F.Y. (Ed.), <u>Recent Developments in Structural Optimization</u>, American Society of Civil Engineers, 1986.

17. Cheng, F.Y. and C.P. Pantelides, "Optimum Seismic Structural Design with Tendon and Mass Damper Controls and Random Process," ASCE <u>Structures Congress V</u>, pp. 40-53, 1986.

18. Cheng, F.Y. and Chang, C.C., "Aseismic Structural Optimization with Safety Criteria and Code Provisions," Proceedings of the U.S.-Asia Conference on Engineering for <u>Mitigating Natural Hazards Damage</u>, Vol. I, pp. D8-1-D8-12, 1987.

19. Cheng, F.Y. and Pantelides, C.P., "Optimal Active Control of Wind Structures Using Instantaneous Algorithm," <u>Vibration Control and Active Vibration Suppression</u>, ASME, pp. 21-28, 1987.

20. Cheng, F.Y., "Response Control Based on Structural Optimization and its Combination with Active Protection," <u>Proceedings of 9th World Conference on Earthquake Engineering</u>, Vol. VIII, pp. 471-476, 1988.

21. Cheng, F.Y. and Juang, D.S., "Assessment of Various Code Provisions Based on Optimum Design of Steel Structures," <u>International Journal of Earthquake Engineering and Structural Dynamics</u>, Vol. 16, pp. 45-61, 1988.

22. Cheng, F.Y. and Juang, D.S., "Recursive Optimization for Seismic Steel Frames," <u>Journal of Structural Engineering</u>, ASCE, Vol. 115, No. 2, pp. 445-466, 1988.

23. Cheng, F.Y. and Pantelides, C.P., <u>Combining Structural Optimization and Structural Control</u>, Technical Report NCEER-88-0006, 1988.

24. Cheng, F.Y., and Chang, C.C., "Optimum Studies of Coefficient Variation of UBC Seismic Forces," <u>Proceedings of 5th International Conference on Structural Safety and Reliability</u>, ASCE, Vol. III, pp. 1895-1902, 1989.

25. Cheng, F.Y., "Optimum Design of Seismic Structures: Methodologies and Results," <u>Proceedings of 4th US NCEER</u>, Vol. 2, pp. 917-927, 1990.

26. Davidson, J.W., Lewis, P., and Hart, G.C., "On Reliability-Based Structural Optimization for Earthquakes", <u>Comput. Struct.</u>, Vol. 12, 99-105, 1980.

27. Furukawa, K. and Furuta, H., "A New Formulation of Optimum Aseismic Design Using Fuzzy Mathematical Programming", <u>8th WCEE Proceedings</u>, Vol. 5, 443-450, 1984.

28. Guerra, O.R. and Esteva L., "Optimization Criteria for the Selection of Ground Motion Isolators," <u>7th WCEE Proceedings</u>, Vol. 8, 163-165, 1980.

29. Kawamura, H., Tani, A., etc., "Optimum Aseismic Design of Buildings Based on Production Rule and Fuzzy Theory," <u>9th WCEE Proceedings</u>, Vol.V, pp. 1019-1024, 1988.

30. Kelly, J.M., "Aseismic Based Isolation," <u>The Shock and Vibration</u> Digest, 14, 17-25, 1982.

31. Kelly, J.M., "Aseismic Based Isolation: Review and Bibliography," <u>Soil Dynamics Earthquake Eng.</u>, Vol. 5 (4), pp. 202-217, 1986.

32. Kilimnik, L. Sh., and Tonoyan, K.A., "Optimal Response Analysis of Multi-Story Frame Structures Under Seismic Excitations," <u>Proceeding of 7th European Conference on Earthquake Engineering</u>, Vol. 3, 501-510, 1982.

33. Kim, S.H. and Wen, Y.K., "Optimization of Structures Under Stochastic Loads," <u>Structural Safety and Reliability</u>, Vol. III, pp. 1871-1878, 1989.

34. Kobori, T., "Active Seismic Response Control," <u>9th WCEE Proceedings</u>, Vol. 8, pp. 435-436, 1988.

35. Liu, S.C., Jang, J.N., and Samali, B., "Control of Coupled Lateral-Torsional Motion of Buildings Under Earthquake Excitation," <u>8th WCEE Proceedings</u>, Vol. 5, 1023-1030, 1984.

36. Pall, A.S. and Marsh, C., "Response of Friction Damped Braced Frames," ASCE, J. ST.D., Vol. 108, 1313-1323, ST6, 1982.

37. Pantelides, C.P. and Cheng, F.Y., "Optimal Placement of Controllers for Seismic Structures," <u>International Journal of Engineering Structures</u>, 1990.

38. Truman, K.Z. and Cheng, F.Y., "Optimum Assessment of Irregular 3-D Seismic Buildings," <u>Journal of Structural Engineering Division</u>, ASCE, Nov., 1990.

39. Soong, T.T., "State-of-the-Art Review: Active Control in Civil Engineering," <u>Journal of Engineering Structures</u>, Vol. 10, pp. 74-84, 1988.

40. Venkayya, V.B., and Cheng, F.Y., "Resizing of Frames Subjected to Ground Motion," <u>Proceedings International Symposium on Earthquake Structural Engineering</u>, (Ed. F.Y. Cheng) 1, 597-612, 1976.

41. Yamada, Y., Iemura, H., etc., "Phase Delayed Active Control of Structures under Random Earthquake Motion," <u>Proc. 4th US NCEER</u>, Vol. 3, pp. 447-456, 1990.

42. Yamada, Y., Furukama, K., Kitajima, K., "Studies of the Effect of Earthquake and Structural Uncertainties on Optimum Aseismic Design of Long Span Suspension Bridges," <u>7th WCEE Proceedings</u>, Vol. 6, 721-728, 1980.

43. Yang, J.N., "Building Critical-Mode Control: Nonstationary Earthquake," <u>Journal of Engineering Mechanics</u>, ASCE, Vol. 109, 1375-1389, 1983.

44. Yang, J.N., Akbarpour, A., and Ghaemmaghami, P., "New Optimal Control Algorithms for Structural Control," <u>Journal of Engineering Mechanics</u>, ASCE, Vol. 113, No. 9, pp. 1369-1386, 1987.

45. Zingone, G., Papia, M. and Russo, G., "Structural Ductility in Reinforced Concrete Buildings Stiffened by Cross Braces Criteria for Dimensional Optimization," <u>7th WCEE Proceedings</u>, Vol. 3, 199-207, 1982.
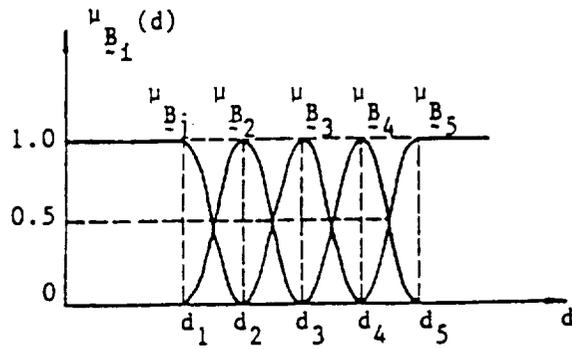
Fig. 4. Membership function of fuzzy
damage grades
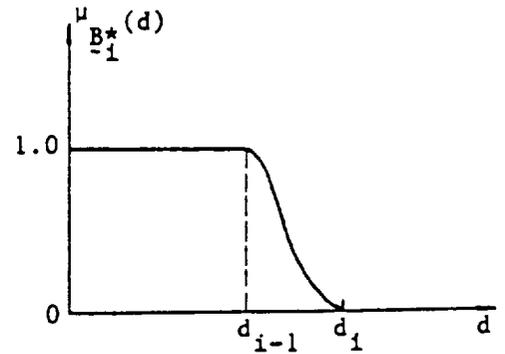
Fig. 5. Membership function
of fuzzy safe regio

Fig. 6. Cross section (I) vs. cost (C_T)

177
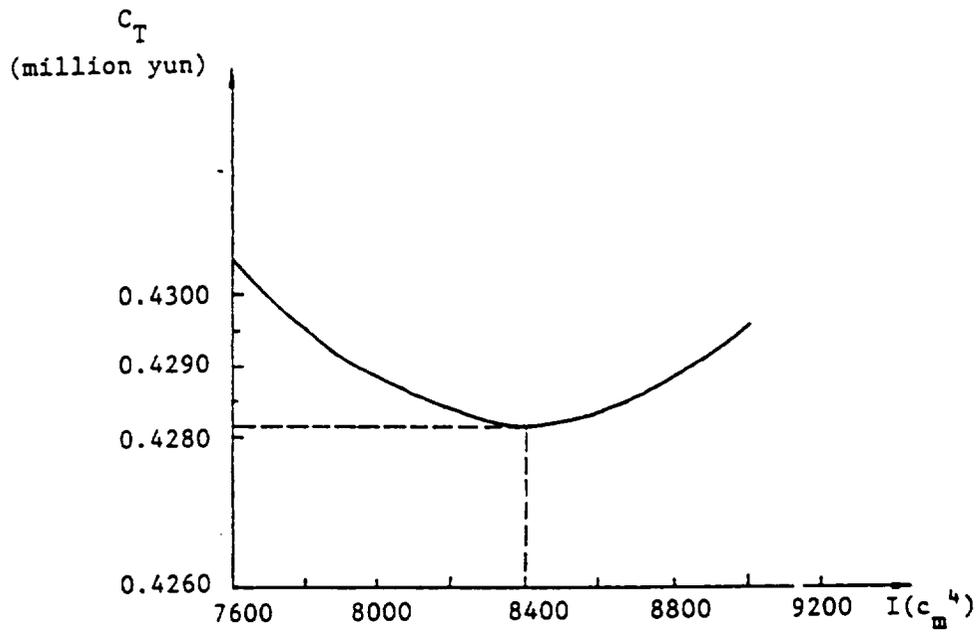
Control/Structure Interaction
Conceptual Design Tool

by
Dr. Hugh C. Briggs
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109

Abstract

The JPL Control/Structure Interaction Program is developing new analytical methods for designing micro-precision spacecraft with controlled structures. One of these, the Conceptual Design Tool, will illustrate innovative new approaches to the integration of multi-disciplinary analysis and design methods. The Tool will be used to demonstrate homogeneity of presentation, uniform data representation across analytical methods and integrated systems modeling. The Tool differs from current "integrated systems" that support design teams most notably in its support for the new CSI multi-disciplinary engineer.

The Design Tool will utilize a 3-dimensional solid model of the spacecraft-under-design as the central data organization metaphor. Various analytical methods, such as finite element structural analysis, control system analysis and mechanical configuration layout, will store and retrieve data from a hierarchical, object oriented data structure that supports assemblies of components with associated data and algorithms. In addition to managing numerical model data, the Tool will assist the designer in organizing, stating and tracking system requirements.

The Design Team

A team oriented model underlies the CSI design process and a brief description will be presented by way of introduction to provide a working model of the Tool's "user community." The principal objective of the design team is to find a design that solves a given problem in some optimal sense. Before candidate designs can be generated, the problem to be solved must be determined and all appropriate constraints identified. The design space, or those solutions that might be considered, must be searched and the performance of feasible designs evaluated. The team may find it has been given several problems to solve or that certain aspects are vague and ill-defined.

Project leadership is frequently split between a program manager and a technical manager, where the technical manager leadsg the design team with the help of a systems engineer. The team works on behalf of stakeholders such as sponsors and benefactors. The problem statement comes, in some form or other, from the stakeholders. The team analyses possibilities, presents alternatives with descriptive data to a "decision maker", and returns to work with the decision and additional data to refine the solution.

The life cycle of a system spans stages such as design, production, operations/maintenance, modification and retirement. While the analytical elements of the design environment could well be used in the investigation of anomolies during the operations phase and certain design features could be used in the modification of a mature system, the present discussion is aimed at the initial design phase. Here, the clean-slate nature of the early design efforts probably requires significantly different designer support functions than the modification and maintenance phases where the system configuration is tightly constrained and the history of the system provides considerable inertia to change.

Even within the design phase, various requirements can be identified. During conceptual design, the team will have only a few members and possibly be just a single system designer. The objective is to capture the intentions of the end user, turn these into requirements and generate one or more candidate designs. Use of analytical methods is minimal since the configuration and component parameters may be poorly known and the design process support might be mostly visual presentation of the configurations. Simple representative models and compilations of broad design rules might be used to quantify performance. At the end of this phase of design, reasonable candidates have been identified and major advantages and shortcomings of each described. User requirements have been negotiated with anticipation of "what is possible" having been traded against "what is necessary."

In the next stage which culminates in a review of the preliminary design, the configuration of the leading candidates are optimized and the uncertainties associated with critical elements are evaluated in some depth. More analysis is possible as full featured system level models can be built and quantitative preformance measures can be evaluated. By

the end of this phase, a single candidate system has been identified and analyzed with little risk remaining in most requirement versus performance areas. The system design team has grown significantly to as many as ten members and representatives of the major subsystem elements have received their initial requirements.

In the remaining period prior to full scale production, detailed analyses of subsystems and components are performed to evaluate the remaining risk elements and to prepare fabrication and assembly details. Since the system configuration is quite stable, high fidelity modeling can be employed to evaluate the exact consequences of subsystem interactions and to prepare guidelines for system level testing.

Advances in the development of design environments for the prelimary design phase will come primarily in the integration of individual technology analyses to support system level analysis and allow efficient trade studies at the system level. State of the art discipline methods are generally not required and model interfacing, data passing, and information management are the key issues. Support for the multi-disciplinary design team members requires formulation of formerly disparate discipline methods in a common symbology, terminology and analytical presentation. It is not until the later detailed analysis and verification stages that in-depth analysis by technical specialists will be required.

On the other hand, considerable development of new technologies is required to support the design team during conceptual design. Rapid preparation of realistic depictions of the system are required to support configuration layout and evaluation. Knowledge bases containing as yet unidentified design rules must be integrated and made available as a consultation service on request or an oversight function with notification on exception. Surrogate subsystem models must be catalogued and available for trial and evaluation. The data manager must maintain several candidate system concepts complete with models, analysis results, and notes. History and audit trails will eventually be required. Finally, a smooth transition to the preliminary design phase must be supported by translation and extrapolation of models, selection of candidates to be carried forward, and selective improvement in model fidelity and system detail.

Additional development is required in the area of requirements capturing, quantification and tracking. The design team must interpret user needs and generate a data base of system requirements and derived requirements. These must be available later for comparison to analytically predicted system performance and for reconciliation of discrepancies. Traceability is very important and some capability beyond simple text documents must be provided.

The Nature of Problem Solving and Synthesis

The construction and application of models can serve two different purposes and the design tool should support both.[1] The traditional use of models is to analyze the performance of a system, typically for the purpose of making a decision about its adequacy or acceptibility. The design tool must at least accomplish this, but with a new capability to analyze multi-disciplinary systems in a team context. The usefulness of a given model or analytical method depends on its accuracy versus its cost. More to the point, in a system design context, the merit of a tool lies in the ability of the team to control the accuracy versus cost trade in order to invest in increased accuracy when required and to avoid unnecessary charges when more approximate results are appropriate.

The second purpose of modeling is to achieve an improved understanding of complex systems as might be required early in the design process when subsystems are being synthesized into system concepts. The tool must provide a more exploratory and investigative style where interaction and quick response are important. In this context relatively little might be known about the system and the designer may be working more to establish a well posed problem. "The purpose of the model lies in the act of its construction and exploration and in the resultant, improved intuition about the system's behavior, essential aspects, and sensitivities."[1]

A general tool must support various world views and approaches to problem solving. No single model of the synthesis methods employed by designers exists and several different approaches can be discerned with little trouble. While a completely general enviornment might be desirable, it is probably impractical. Consider the fundamental differences between declarative languages such as Prolog and procedural languages such as Fortran. Therefore, an understanding of the major elements in approaches to design can indicate which functions should be implemented and made available for trial and proof testing.

An Analysis of Design Tasks

The activities accomplished by a designer span a broad range in terms of level of abstraction and technical knowledge. By understanding the tasks, attitudes and activities of the designer, the feature set of the design tool might be chosen to include a rich set of supporting functions. It remains to be seen how automation will change the way designers work and what changes this will bring to the evolution of such design tools.

Tasks can be described by the level of abstraction required to comprehend the breadth of the task and the understanding of the actions involved.[2] For example, the first three levels represent work that involves relatively concrete types of thinking. "Perceptual-motor concrete" is a mode of work that involves direct perceptual contact with the physical output. The second level, "imaginal concrete," requires the use of imagination in constructing a project, but deals with projects for which the final output can be visualized in concrete terms. The third level is "imaginal scanning" which involves tasks where it is impossible to comprehend an entire area of responsibility at once, although the whole can still be mentally scanned, one piece at a time.

There is a significant change in the level of abstraction between levels three and four since there is a change from the concrete to the abstract mode of thought and work. "At level four, neither the output nor the project can be foreseen in concrete terms, even by imaginal scanning. The project cannot be completely constructed. It remains a combination of a conscience subjective picture, incomplete in itself, whose specific total form and content are unconsciously intuitively sensed but cannot quite be consciously grasped."[2] The fifth level is based upon the intuitive theories the individual has developed from his experience. It seems that the art supplied by senior spacecraft designers involves operations at these higher levels.

Six activities can be identified within the design task that might be impacted by the design tool. Many of these tasks are currently accomplished mentally, particularly in the earlier design phases, and cannot be easily shared or transferred. A goal of this tool development is to facilitate both the shared construction of the design and the depiction of the design so that it might be transferred.

Table 1. Complexity of Design Activities

Activity	Level of Abstraction	Task Examples
Requirements Analysis	low	data gathering and fact finding
	imaginal concrete	synthesis of requirements
Configuration Synthesis	intuitive	choosing among alternatives
	imaginal scanning	assembly of system configuration
Model Building	concrete	preparation of model data
	imaginal	planning analyses
Analysis	imaginal scanning	assimilating system performance
	concrete	executing analysis programs
Decision Making	highest intuitive	choosing "best" alternatives
		preparing advocacy strategies
Reporting	moderate	building documentation
	high	responding to criticism

An alternative view considers the preferences people have for assimilating data and making decisions. For example, approximately 75% of the general population prefer to study data in quantity before identifying structure and drawing conclusions or making predictions.[3] Such people collect data and view the problem from many sides before finding an understanding of the structure and nature of the underlying phenomina. Others generalize more readily, hypothesizing structure and looking for large scale trends. Further data is then collected to see if they support the generalizations.

Although large amounts of data exists to describe these and other psychological preferences in the general population, no data has been collected on the special subset made of spacecraft designers and design team members. To the contrary, it is easy to find examples of the diverse qualities and habits exhibited in the aerospace design community.

A few general requirements might be proposed based upon these considerations. Beyond support for interactive multi-disciplinary analysis, various forms of data handling, analysis and viewing will be needed. For example, data collection and querie systems will be needed to support those who find structure in data. Testing of ideas and possibilities will be required for those who generalize and then verify against the data. Various levels of data aggregation will be required as questions move from the concrete of element and component analysis to the general questions of overall spacecraft success.

The Design Tool

A few central themes are embedded in the implementation of the Design Tool that differentiate it from existing tools. These lie in the unified formulation of the analytical methods and the structure of the computer program itself.

The following sections describe the visual presentation of the tool, management of internal data, information analysis features, and the modeling methods for structural dynamics, controls, optics, and thermal analysis.

Visual Presentation

The multi-faceted nature of system design and the "fire fighting" character of system level trade studies suggest the design team can best be supported by environments that provide multiple, parallel access paths to the design tool elements. For example, model building and analysis should be supported with simultaneous access to a documentation and note taking facility. Workstation computer systems with windowed presentation managers provide this capability by allowing the user to activate any tool method in a window and directing his attention or focus to the needed window. The manipulation of the data base by the underlying tools must be coordinated in a consistent, non-disruptive manner.

A principal tenet of the present effort is that a realistic depiction of the system be available at all times. A 3-D solids model of the system is to be shown in a window which has controls for rotating and otherwise manipulating the presentation. Results of analysis, trades and configuration changes are to be fed back visually through interactive alteration of the solid model. Selection of components, element groups, and system features should be available by pointing into the 3-D display and the selections made available as inputs to other tools in the environment.

Analytical Methods

Traditional discipline methods for modeling and analysis can readily be employed in system design if care is taken to cast them into a common framework so that efficient integration can be accomplished. In general, system level analyses do not require in-depth analysis or non-linear methods and models are frequently large collections of simple elements. This necessitates efficient data storage and careful attention to implementation of numerical algorithms but such developments have largely been made within each discipline.

Spacecraft design requires several analytical methods, most notably structural dynamics and controls. Certain spacecraft require other methods in addition, such as optics for space interferometers or large space telescopes. A common formulation and notation is required to simplify usage by the multi-disciplinary design team and is readily achievable for most constituent methods. The following briefly summarizes several examples.

Structural dynamics are well captured by finite element methods and most spacecraft systems require only simple 3-D truss and frame analysis. The geometry is defined by nodes which are points in 3-space that might have up to three translation and three rotational degrees of freedom. The structural members are described by connectivity lists of nodes and a property list containing parameters such as thickness and material properties. The structural dynamics equations can be stated simply in terms of the system matrices and nodal degrees of freedom as follows.

$$[M]^* \{\ddot{d}\} + [D]^* \{\dot{d}\} + [K]^* \{d\} = \{F\}$$

The vector of forces, $\{F\}$, might include external forces acting on the system such as disturbances, vibrations of on-board equipment, gravity, and forces caused by a control system.

Most methods for the modeling of controlled structures are based upon a description of the system to be controlled (known as the plant) which utilizes first order ordinary differential equations. The degrees of freedom of the plant are collected in a state vector, $\{x\}$, and the forces provided by the controller are in $\{u\}$ so that

$$\{\dot{x}\} = [A]^* \{x\} + [B]^* \{u\}$$

In certain situations, the states themselves may be directly measured, but as is common in controlled structures, the sensed variables $\{y\}$ are not the states but are related to the states by

$$\{y\} = [C]^* \{x\}$$

The control law will be implemented in a controller that coexists with the structure, taking measurements $\{y\}$ and computing commands $\{u\}$. Typical linear control laws compute controller commands based upon constant gains via

$$\{u\} = -[G]^* \{y\}$$

When the plant is modeled by finite elements, the structural dynamic equations can be cast into the first order form used in control analysis by identifying the structural displacements and velocities as the states of the plant. If, instead, a modal model reduction has been applied, the states are the modal amplitudes and velocities. Let $\{x\} = [d \ \dot{d}]^T$. Then the structural dynamics equations become

$$\{\dot{x}\} = \begin{bmatrix} 0 & I \\ M^{-1}K & M^{-1}D \end{bmatrix} * \{x\} + \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix} * [B]* \{u\} + \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix} * \{F\}$$

With the application of the linear control law, the closed loop equations become

$$\{\dot{x}\} = \left[\begin{bmatrix} 0 & I \\ M^{-1}K & M^{-1}D \end{bmatrix} - \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix} *B*G*C \right] * \{x\} + \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix} * \{F\}$$

Other observations that describe the system performance might be constructed from the states by

$$\{z\} = [C']* \{x\}$$

These might include optical beam positions, motions of mirror attachment points, or even the spacecraft line of sight. Optical elements can be used to describe the transfer of light paths through the spacecraft and the image perturbations that result from spacecraft dynamic excitation.

Optical elements are described by the coordinates of the intersection point of the nominal ray and a list of optical parameters such as principal axis direction, focal length and eccentricity. The input ray describes the deviations in terms of an input offset d_i, an input orientation i, and a path length differential dL_i. The departing ray is described by the offset d_o, direction r, and path length differential dL_o. With these degrees of freedom, the element transformation matrix $[t_e]$ is

$$\begin{Bmatrix} d_o \\ r \\ dL_o \end{Bmatrix} = [t_e]* \begin{Bmatrix} d_i \\ i \\ dL_i \end{Bmatrix} \quad ; \quad [t_e] = \begin{bmatrix} \dfrac{dd_o}{dd_i} & \dfrac{dd_o}{di} & 0 \\ \dfrac{dr}{dd_i} & \dfrac{dr}{di} & 0 \\ \dfrac{ddL}{dd_i} & \dfrac{ddL}{di} & 1 \end{bmatrix}$$

The components of this transformation matrix are based upon the physics of beam reflection from a general reflective surface.[4] In general, all degrees of freedom between element 0 which might be an input and element n which might be an output, can be eliminated by the compaction

$$\begin{Bmatrix} d_o \\ r \end{Bmatrix}^n = [t]^{n*} \cdots *[t]^{2*} [t]^{1*} \begin{Bmatrix} d_o \\ r \end{Bmatrix}^0$$

The resulting matrix $[t]$ provides the desired observation matrix $[C']$ that describes the perturbed state of the beam.[5]

Another discipline that can readily be included in the common formulation is thermal analysis of the spacecraft structure. To describe distortions and misalignments caused by temperature changes in structural elements, thermally induced strains can be considered to cause additional loads $\{F\}$ which deform the structure. To retain a simple, linear analysis, dynamics properties and heat transfer properties must be considered to be independent of temperature, an assumption that is reasonably valid for small temperature excursions. In this case, the degrees of freedom are nodal temperatures T and the heat conduction equations are

$$[C] * \{\dot{T}\} + [K] * \{T\} = \{Q\}$$

where $[C]$ are heat capacities, $[K]$ are conductivity coefficients and $\{Q\}$ are the externally imposed heat loads. With the nodal temperatures, the thermal structural loads can be computed from

$$\{F\} = [\alpha A E]\{T\}$$

where α is the coefficient of thermal expansion, A is the element area, and E is the element modulus.[6]

Extensions to include radiative transport mechanisms, as is required for the direct calculation of the heat load $\{Q\}$ must be developed in light of the non-linear nature of the radiation effects.[7] When this problem is combined with the structure heat conduction problem, the coupled problem is non-linear since the energies are proportional to T^4. Since this involves the absolute temperature, one possible approximation is to factor T^3 into the heat transfer coefficient and restrict the relative temperature variations to ranges that produce error comparable to other sources. Furthermore, the calculation of the required view factors involves extensive calculations in 3-D geometry and may not be cost effective in

trades involving radical changes in system configuration. These linearizations and approximations are appropriate for preliminary design analyses and have been used in the past.

Data Analysis and Synthesis

The ability to collect, organize and evaluate data efficiently is critical to the synthesis of ideas and their evaluation. For example, the structural performance of the candidate spacecraft must be catalogued and evaluations summarized succinctly in order to comprehend trends versus design variables, to capture phenomina related to resource allocation and expenditure trades between subsystems and to create audit trails supporting choices and decisions. Analysis of meaningful models in the necessary miriad of cases can generate data in several forms and large volumes. Not only must traditional data management techniques be integrated into the analytical environment but implementations of new juxtapositions of technologies must be evaluated.

Analysis of data bases formed from regular tables of case data can be supported with structured query language (SQL) investigations of relational data bases.[8] A typical area might be to collect in a table the maximum stress in structural members for various cases and geometric data such as element locations in another table. Requests such as

select	member#, maxstress
from	stresstable, locations
where	maxstress > somelimit
and	is_in_rt_boom (location) = TRUE

will generate the requested lists.

Use of such data bases requires that analysis data be cast into relational tables of "tuples", eg. {member#, case#, maxstress}. Constructing such tables can be difficult in system trades when differing configurations lead to different types of structural elements or non-equivalent cases. A more generalized data structure and inquiry process is required to support the investigation and exploratory evaluations that accompany system synthesis.

Hierarchical and object oriented data bases allow a general knowledge data base to be constructed by tagging objects with properties and relationships. For example, truss elements might have a structure such as

name	truss element
id number	1012
matl id	3
connectionA	22
connectionB	103
maxstress	1324.5

Then structural assemblies can be viewed as objects with properties, one of which is a list of elements.

name	left boom
element list	{..., 1012, ...}
instrument list	{...}

Inquiries similar to the SQL example can be answered by searching the object data base for objects with property values that satisfy the conditions in the clause.

The data associated with a design problem can be stored in a base and viewed as required as a relational data base and queried via SQL or as an object data base and queried based upon descriptive properties.[8] This data can also be viewed in other forms, such as logic or production rules and queried in Prolog-like languages. The implementation of such a versatile data manipulation and analysis facility would probably store data in an efficient internal representation but provide each of the several (SQL, object, rules, ...) views at the user's discretion.

Data Storage

While the most common data type in traditional spacecraft analysis is the numeric array, a more general structure can facilitate tagging, tracking, documenting and auditing of analyses. For example, a mixed type record could store case results:

case_name	"vertical pitch maneuver"
configuration	"pre-deployment"
pitch_rate	32.5
case_id	13
maxstress_list	{..., ..., ...}

A generalization of this is contained in the object oriented approach to data representation. While current object oriented programming practice includes many features such as encapsulation, object specific methods, it may suffice to simply support user defined mixed type records and methods to associate such records into hierarchical sets.

For spacecraft design, considerable attention to efficient handling of numeric data will be required. For the most part, internal representations of arrays can be hidden from the user as can the type casting issues associated with mixed mode arithmetic. Implementation of sparse arrays, possibly in several formats, is a necessity. Structural dynamics, heat conduction, and certain control algorithms utilize sparsely populated arrays and prior experience in structural dynamics suggest a large payoff here. Certain linear algebra routines and standard analysis practices destroy the sparse matrix property but some work arounds are possible. For example, in control system design, eigen analysis typically results in the calculation of all system eigenvectors. If standard practices of structural dynamics can be adapted, only a few eigenvectors in the band of interest could be calculated, saving calculation time and storage.

Thus, the data manager must support efficient storage for named objects with a general structure. Functions for creation, deletion and copying must be provided, as well as merging, partitioning and structured assembly. Arithmetic operations must be supplied for the base numeric types and a facility for user defined operations on general types will be required. Advanced systems must address archiving, backup, persistence across interrupted sessions and auditing.

Summary

The Design Tool will support spacecraft design teams with an integrated set of analytical tools, a versatile data manager, full-time 3-D solid model views, and a multiple thread presentation manager. Tasks with a range of complexity from data input to analysis to system synthesis are supported through a general data manager and query system. The requirements for the Design Tool have been developed and an initial implementation begun although the completion of the first prototype has been delayed due to funding limitations.

Acknowledgement

References

1. Kreutzer, W., "System Simulation Programming Styles and Languages," Addison Wesley, 1986.

2. Rockart, J.F., and DeLong, D.W., "Executive Support Systems", Dow Jones-Irwin, 1988.

3. Kiersey, D., and Bates, M., "Please Understand Me, Character and Temperament Types," 4 Ed., Prometheus Nemesis Book Co., 1984.

4. Redding, D.C., and Breckinridge, W., "Optical Modeling for Dynamics and Control Analysis," to be presented at AIAA 1990 Guidance and Control Conference, Paper 90-3383, Portland, OR, 20-22 Aug 1990.

5. Briggs, H.C., Redding, D.C., and Ih, C.C., "Integrated Control/Structure/Optics Dynamic Performance Modeling of a Segmented Reflector Telescope", Proceedings of the Twenty First Annual Pittsburgh Conference on Modeling and Simulation, 3-4 May, 1990.

6. Cook, R.D., "Concepts and Applications of Finite Element Analysis," 2 Ed., John Wiley and Sons, 1981.

7. Holman, J.P., "Heat Transfer", 2 Ed., McGraw-Hill, 1968.

8. Parsaye, K., Chignell, M., Khoshafian, S., and Wang, H., "Intelligent Data Bases," John Wiley and Sons, 1989.

GENERAL PURPOSE OPTIMIZATION SOFTWARE
FOR ENGINEERING DESIGN

by

G. N. Vanderplaats, President
VMA Engineering
5960 Mandarin Ave., Suite F
Goleta, CA 93117

INTRODUCTION

The author has developed several general purpose optimization programs over the past twenty years. The earlier programs were developed as research codes and served that purpose reasonably well. However, in taking the formal step from research to industrial application programs, several important lessons have been learned. Among these are the importance of clear documentation, immediate user support, and consistent maintenance. Most important, has been the issue of providing software that gives a good, or at least acceptable, design at minimum computational cost. Here, the basic issues developing optimization software for industrial applications are outlined and issues of convergence rate, reliability and relative minima are discussed. Considerable feedback has been received from users, and new software is being developed to respond to identified needs. The basic capabilities of this software are outlined. A major motivation for the development of commercial grade software is ease of use and flexibility, and these issues are discussed with reference to general multidisciplinary applications. It is concluded that design productivity can be significantly enhanced by the more widespread use of optimization as an everyday design tool.

The CONMIN program (1) was written in 1972 as a general purpose optimization code, and it is still widely used. This code is based on the Method of Feasible Directions and was documented via a NASA Technical Memorandum. It was developed as a research program at NASA Ames Research Center. While numerous enhancements were made over the years, there was no formal maintenance or upgrade mechanism. Consequently, the user does not even know what version of the program he may be using. In 1984, the ADS program (2) was released. This code was also developed in a research environment under the sponsorship of NASA Langley Research Center. ADS contains numerous Strategy, Optimizer and One-dimensional Search options for a total of about 100 different overall algorithms possible. A private User's Group was created to maintain and enhance the program, and this existed until 1988. ADS proved to be a very useful academic aid since students could experiment with a wide variety of methods. However, this generality, as well as the fact that many of the algorithms contained in ADS are inefficient or otherwise obsolete, made the program less attractive to the industrial user who wishes to have a simple and reliable interface. Thus, the wide range of capabilities of ADS were, in many ways, a detriment to efficient use of the program. Also, it was supported only on a part time consulting basis.

Recognizing the need for a simple interface with optimization, in 1976 the COPES/CONMIN driver program was created. This main program made it much simpler to interface the user's analysis program with CONMIN, since he no longer had to provide a main program to create the optimization task in standard form. The design variables, objective and constraint functions were identified via input data. Also, COPES provided additional features for parametric studies and optimization using approximations. This last feature allows the user to perform optimization based on experimental data. This code was developed as a research program and was not formally documented until it was converted to work with ADS in 1984 (3).

In 1986, the DOT optimization program was released as a commercially supported general purpose code (4). Development of this program was heavily influenced by experience gained with CONMIN and ADS. Version 1 contained only one optimization algorithm, the Modified Method of Feasible Directions (5), and was small enough to be used on a personal computer (this version was called Micro-DOT). DOT is written in FORTRAN 77, and is more formally documented and supported. COPES was modified to become COPES/DOT. Version 2 of DOT contains a second optimization algorithm, Sequential Linear Programming. It is noteworthy that the algorithms contained in DOT are not considered to be "robust" by the theoreticians in optimization, who today prefer the Sequential Quadratic Programming algorithm. However, these algorithms are used because they retain an acceptable level of efficiency and reliability over a wide spectrum of applications. This is a key issue in development of software for widespread commercial use, where we must choose between what is best in theory and what is best in practice.

USER ISSUES

Today, most optimization applications in industry are still in the research departments, where the users are reasonably expert in the underlying theory. However, there has been a consistent (long overdue) trend to move this technology from the research to the production environment. This is more pronounced in Asia, where the distinction between research and applications engineers is not usually clear and there is a greater interest in new technology which will improve product quality, even when that technology cannot be shown in advance to generate a short term profit.

Based on experience with this growing group of users, several key issues can be identified which are important to the industry user. These issues (this is not a conclusive list, but is representative) can generally be ranked as follows:

1. Immediate and competent response to technical problems must be provided. This includes competence in general engineering design, since a critical issue is often problem formulation techniques.

2. Ease of use by practitioners who have little or no background in optimization theory is essential.

3. The software must reliably achieve a reasonable design improvement. A precise optimum is recognized as being of little value since the design will change anyway.

4. The software must be efficient in terms of the number of analyses needed to reach the optimum. Gradient information will almost never be provided analytically, but will instead be calculated by finite difference (this must be available as part of the optimization software).

5. Clear and concise documentation must be available.

6. Continual maintenance and enhancements must be assured.

7. Source code must be available (even though nobody reads or changes it)

8. The software must be available at reasonable cost on a large variety of computers.

These issues are much as would be expected, although their order may be subject to debate. Most importantly, few users of optimization who are applying it to real design tasks have much interest in what algorithm is contained in the program. They only care that it works! Also, while researchers worry about relative minima, practitioners usually recognize this as a fact of life which exists whether or not optimization is used. While they would like to have software that finds the absolute minimum, they quickly understand the futility of this and are willing to run the optimization from several reasonable starting points in search of the best solution. Finally, engineering design is a study in exceptions, rather than rules. Optimization software must be flexible enough to accommodate this fact.

THE DOC PROGRAM

In response to input from the user community, as well as the fact that the COPES program was developed as a research tool without a consistent format, a new interface program called DOC (Design Optimization Control program) is being developed in order to enhance the usefulness of optimization. While DOC uses the same philosophy as COPES, where the problem is defined via user input, the form of the input has changed from a formatted type to a more readable mnemonics interface. DOC includes the capabilities of general optimization, parametric studies and optimization using approximations, but is formally documented and supported. Also, new capabilities are included and others can more easily be added in the future. Three key capabilities that have repeatedly been identified by the users as desirable are multiobjective optimization, synthetic functions, and optimization with discrete variables. Because these topics are still active research areas, there is little consensus as to which approach is best. Therefore, this decision is based on experience with various methods and the necessary reliability of the final code.

Multiobjective Optimization

The usual comment made when first considering the use of optimization is that most design tasks are represented by multiple objective functions. This may include weight and cost minimization, range and performance maximization and a host of others. While it may be argued that this is a naive approach to optimization, since such a problem statement will have no unique solution, it is nonetheless a real concern which must be addressed.

Numerous methods have been proposed for multiobjective optimization, but the most acceptable appears to be compromise programming (6), where the single objective presented to the optimizer is defined as;

$$F = \left\{ \sum_{j=1}^{J} w_j \left[\frac{R_j - R_j^*}{R_j^{worst} - R_j^*} \right]^2 \right\}^{1/2}$$

where R_j is the jth objective function, R_j^* is the jth "target" objective value and R_j^{worst} is the worst known value of the jth objective function. The parameter, w_j, is a weighting factor defining the relative importance of this objective. It is common to perform a single objective optimization to determine the target value,

R_j^*. This provides a good baseline design which identifies the best that can be achieved for this objective without regard for the others. The value of R_j^{worst} is usually chosen as the best known value of this objective prior to optimization.

Synthetic Functions

It is common that the user wishes to optimize or constrain a function which is not directly supplied by the analysis program. Or he may wish to use parameters as design variables that are not normally input to the analysis. For example, he may require that the sum of two responses be less than the value of a third response, or he may wish to treat physical dimensions of a frame structure as design variables, although the input to the analysis is section properties, which are functions of these variables. In many cases, he can simply modify the analysis program to provide this information in the appropriate format, but often he is not able to change the analysis program or wishes to experiment with new ideas before making changes. The solution then is to provide the capability to use "synthetic" functions for optimization (7). Typically, the user might input (as data) an equation in a form similar to a FORTRAN FUNCTION statement.

For example, if it is desired to treat the width, B, and height, H, of a rectangular beam as design variables, but the analysis program input is the section properties, this can be accomodated using synthetic definitions of the properties. Thus, the area moment of inertia would be defined as

I(B,H) = B*H**3/12.0

A stress recovery location at the corner of the section may be defined as

R(B,H) = 0.5*SQRT(B**2+H**2)

Similarly, the allowable buckling stress in a rod may be defined as

$$\sigma_b = \frac{-40*A*E}{L^2}$$

Since this must be less than the actual stress, the synthetic equation would be

G(A,E,L,SIG) = (-40.0*A*E/L - SIG)/10000.0

where 10000.0 is a normalization factor and SIG is the calculated stress. Now this equation is identified as a constraint to be non-positive in the optimization process.

By providing this capability, the user has a great deal of flexibility in defining the optimization task and experimenting with various concepts, without making them a permanent part of his analysis program.

Discrete Variables

Discrete variable optimization is still an active research area, particularly as it applies to nonlinear optimization in engineering. However, it is an important issue in such applications as composite structure design and where parts must be selected from a set of available components (such as steel structure sections).

Perhaps the most common approach to discrete variable optimization is the branch and bound method (8). Here, a single variable is held at a discrete value and optimization is performed as a continuous problem with respect to all other variables. This variable is then increased to it's next discrete value and the optimization is repeated with respect to the other variables. If the optimum is greater than before, this "branch" is cut off and the variable is set to it's next lower value and the process is repeated. If an improvement is found, the search is continued in this direction until no improvement can be found. Then this variable is held fixed and the process is repeated for the next discrete variable, until all discrete variables have been examined. One disadvantage to this approach is that a multitude of nonlinear optimization tasks must be performed, which is a very expensive process. More importantly, this method was developed for linear problems, but has been applied to nonlinear problems. However, a basic assumption of the method has been ignored in making this step. This is that the problem must be separable, a condition that is seldom true for nonlinear engineering design. The result is that the true discrete optimum can seldom be achieved, even for the most simple cases.

An alternative approach is to first create a separable approximation to the real problem and then solve this as a discrete variable problem using duality theory (9). This has the advantage that is quite efficient, but retains the disadvantage of ignoring coupling among the design variables. Again, a true optimum cannot be guaranteed. However, this method appears to be reasonable for applications such as structural design, where a high quality separable approximation can be created.

The best compromise for general applications is to first solve the problem as a continuous variable task. This provides a lower bound on the discrete solution. Now the problem is linearized and solved as a discrete variable problem. One approach to this is to solve the linearized problem using discrete variable linear programming with branch and bound methods (10). This has been found to be efficient, but to oscillate about the optimum, due to coupling among the variables. An alternative is to use "convex" (combined direct and reciprocal) approximations (9) and duality theory to obtain the final discrete solution. This can also create oscillations about the optimum that must be detected, but it provides the necessary efficiency and provides a near optimum design, especially if the discrete values are closely spaced. The net effect is that a discrete variable optimization capability can be provided in a general purpose optimization code, but that the user must be aware of it's limitations.

SUMMARY

As anyone who has worked in optimization theory and applications will attest, it is desirable to have a fundamental understanding of the process in order to achieve reliable results. In special cases, such as structural optimization, we have sufficient knowledge and experience to create highly efficient design programs. However, the vast majority of potential applications are in areas where we know little about the mathematical structure of the problem, but recognize that optimization has a great deal to offer. In these cases, the key issue is our ability to provide software that efficiently gives some design improvement.

The step from research and academic use of optimization to industrial use as a design tool is a major one. The practitioner has little interest in the actual algorithm used, but is very knowledgeable about the practicalities of design. In the quest to make optimization an everyday design tool, the concerns of the user cannot be ignored. Many of these concerns are related to ease of use, documentation and support. However, a more fundamental issue is that of providing methods and software that are responsive to the real and perceived needs of the design engineer. This is the fundamental challenge in moving optimization from the research environment to the design engineer's desk.

REFERENCES

1. Vanderplaats, G. N., "CONMIN - A FORTRAN Program for Constrained Function Minimization," NASA TM X-62,282, Aug. 1973

2. Vanderplaats, G. N. and Sugimoto, H., "A General-Purpose Optimization Program for Engineering Design," Internal Journal of Computers and Structures, Vol. 24, No. 1, 1986, pp. 13-21.

3. Vanderplaats, G. N., "COPES/ADS - A FORTRAN Program for Engineering Synthesis Using the ADS Optimization Program," Department of Mechanical Engineering Report, Naval Postgraduate School, Monterey, CA, Oct. 1984.

4. Hansen, S. R., Vanderplaats, G. N. and Miura, H., "Micro-DOT Users Manual," Engineering Design Optimization, Inc., Santa Barbara, CA, 1986.

5. Vanderplaats, G. N., "An Efficient Feasible Directions Algorithm for Design Synthesis," AIAA Journal, Vol. 22, No. 10, Oct. 1984.

6. Duckstein, L., "Multiobjective Optimization in Structural Design," New Directions in Optimum Structural Design," Ed. E. Atrek, R. Gallagher, K. M. Ragsdell and O. C. Zienkiewicz, John Wiley & Sons, 1984, pp. 459-481.

7. Vanderplaats, G. N., Miura, H., Cai, H. D., and Hansen, S. R., "Structural Optimization using Synthetic Functions," Proc. 30th Structures, Structural Dynamics and Materials Conference, Mobile, AL, April 3-5, 1989, AIAA CP89-1222.

8. Olsen, G. and Vanderplaats, G. N., "A Method for Nonlinear Optimization with Discrete Variables," AIAA Journal, Vol. 27, No. 3, March 1989, pp. 352-358.

9. Dakin, R. J., "A Tree Search Algorithm for Mixed Integer Programming Problems," Computer Journal, Vol. 8, 1965, pp. 250-255.

10. Fleury, C. and Braibant, V., "Structural Optimization - A New Dual Method Using Mixed Variables," Int. Journal for Numerical Methods in Engineering, Vol. 23, 1986, pp. 409-428.

TROJID - A Portable Software Package for Upper-Stage Trajectory Optimization

Steven M. Hammes
The Aerospace Corporation
El Segundo, CA

Abstract

Performance optimization for upper-stage exoatmospheric vehicles often is performed within the framework of a full capability trajectory simulation package requiring either a large mainframe computer or powerful workstation. Since these software packages tend to include capabilities providing for high-fidelity boost and reentry simulations, the programs usually are quite large and not very portable. The program TROJID is an attempt to provide an environment for the optimization of upper-stage trajectories within a small package capable of being run on a standard desktop microcomputer. Utilizing a state-of-the-art nonlinear programming algorithm and a trajectory simulator implementing impulsive burns and an analytic coast phase propagator, TROJID is capable of producing trajectories for optimal multi-burn upper-stage orbit transfers. The package has been designed to allow full generality in definition of both the trajectory simulator and the parameter optimization problem.

Introduction

Performance optimization is an important tool in the design of space vehicle missions. The high cost of transporting a payload into space demands that missions be designed to utilize all of the capability within the space vehicles. Optimizing an ascent may allow extra spacecraft propellant to be loaded, thus increasing a spacecraft's useable life. Similarly, optimizing the maneuvers required to move a spacecraft from one orbit to another also may increase the spacecraft's lifetime.

A number of general-purpose trajectory simulation packages exist which contain parameter optimization capability. Two state-of-the-art packages are the Generalized Trajectory Simulation (GTS)[1] System and the Program to Optimize Simulated Trajectories (POST)[2]. Each of these programs contains a full capability trajectory simulation package capable of high-fidelity simulations including atmospheric drag, winds, vehicle body attitude, and integration of the equations of motion for powered flight.

In contrast to these large trajectory simulation software packages, the program TROJID (TRajectory Optimization with J_2 and Impulsive Delta-v's) is an attempt to provide an environment for the optimization of upper-stage exoatmospheric trajectories within a portable package capable of being run on a standard desktop microcomputer. The package contains a state-of-the-art nonlinear programming algorithm, NLP2[3,4], along with a trajectory simulator implementing impulsive burns and analytic coasts. The analytic coast phase propagator assumes gravitational acceleration including the J_2 zonal harmonic, suitable for the simulation of multi-burn upper-stage orbit transfers.

Overview

TROJID is a PC-based software package designed to optimize multi-burn exoatmospheric orbit transfers. Such transfers commonly occur when a payload is injected into a geosynchronous mission orbit or hyperbolic escape orbit from a low-earth park orbit, or when a satellite is maneuvered from one orbit to another.

The code is composed of three main elements, as shown in Figure 1. The elements are a system level driver which defines the problem to be solved and interfaces with the user, a robust nonlinear programming algorithm capable of solving constrained optimization problems, and a trajectory simulation package used as a function generator for the optimization routines.
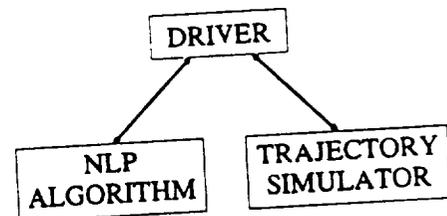

Figure 1 Overall Program Flow

When a new application is designed within TROJID, the system level driver is modified to define the new op-

timization problem and trajectory simulation. Only the driver is modified and recompiled; the optimizer and simulator are maintained in object libraries and linked with the driver to create an executable program.

User-modified input files can be used to initialize specific problems once a new application has been designed and compiled. For example, a basic multi-stage vehicle may be defined within the system level driver, with actual weights and propulsion characteristics being read in from an input file at run time. In addition, definition of the optimization problem can be set up in such a way as to allow an input file to be used to choose the variables, constraints, and objective function for a given problem. Thus, if a new application is designed with care, an entire range of problems may be solved without the need to recompile any code.

The bulk of the code for TROJID is written in standard FORTRAN 77. In addition to the FORTRAN, a few routines used for the DOS run-time screen interface are written in C. Since all but the screen driver routines are written in FORTRAN, TROJID can be hosted readily on any machine with a FORTRAN compiler, although this would imply either running in a purely batch mode with no user feedback during execution, or writing a screen interface for the new environment. In either case, the main elements of TROJID would carry over directly.

The Optimization Algorithm

The optimization algorithm utilized in TROJID is NLP2, a projected gradient algorithm developed at The Aerospace Corporation. The purpose of the algorithm is to find the vector x which minimizes the objective function

$$f(x)$$

subject to the equality constraints

$$c_i(x) = 0$$

for $i = 1, \ldots, m_e$ and the inequality constraints

$$c_j(x) \geq 0$$

for $j = (m_e + 1), \ldots, (m_e + m_l)$ where the total number of constraints is $m = m_e + m_l$. The program has options to use a quasi-Newton method with recursive Hessian updates or a Newton method with finite difference Hessians.

The algorithm begins with a constraint satisfaction phase to find an initial feasible point. Once a feasible point is found, the algorithm switches to solving a series of equality-constrained optimization problems. The optimization problems are solved by using an orthogonal decomposition of the variable space, followed by a separation of the variables into two sets, one of which is used for eliminating the active constraints, the other of which

is used to minimize the objective function restricted to the active constraint surface. If a previously satisfied inequality constraint is reached, the minimization process is terminated and restarted with the new constraint added to the active set. When the minimum is reached for an equality constrained subproblem, the Lagrange multipliers are examined to determine whether any of the currently active inequality constraints can be released from the basis. Once a subproblem is solved and no inequality constraints are added to or deleted from the basis, the algorithm terminates at a local solution.

After a solution has been reached, a postoptimality analysis operator is called[5,6]. This operator serves several purposes. First, projected gradient, Hessian, and eigenvalue data can be calculated in both scaled and unscaled quantities in order to help determine how well NLP2 has converged. Second, Jacobian and Hessian conditioning can be calculated in both scaled and unscaled quantities in order to assist in choosing good scale factors for the problem. These scale factors can be used to improve algorithm performance on other similar problems. Third, sensitivity data can be calculated in order to provide information on partial derivatives at the solution. This information includes partial derivatives of the objective function with respect to changes in constraint values, partial derivatives of the variables with respect to changes in the constraints to maintain feasibility, and second partial derivatives of the objective function with respect to changes in the variables while maintaining feasibility.

The Trajectory Simulator

The trajectory simulator in TROJID is designed to propagate a multi-burn exoatmospheric trajectory beginning with an initial state vector and utilizing a set of coasts and instantaneous velocity impulses (delta-v's), as shown in Figure 2. The simulator is used as the function generator, and any real parameter defining the trajectory can be used as an independent variable in the optimization problem. Typically, parameters such as coast durations, delta-v magnitudes, and delta-v pointing angles are used as variables. Other variables could include vehicle initialization time, initial orbital elements, and vehicle mass and propulsion characteristics.

The initial state vector can be input either as position and velocity relative to the earth or via orbital elements in inertial space. In either case, the first step in the simulator is to translate the state vector into cartesian position and velocity vectors, $\vec{R}$ and $\vec{V}$, within an earth-centered inertial (ECI) reference frame. If desired, a time initialization model may be called in order to align the Greenwich meridian for an input time epoch. The initial ECI state vector is saved before proceeding with the trajectory propagation.

After state initialization, the simulator begins executing n coast-burn pairs, where n is the number of burns being utilized in the simulation. ECI state vectors are saved twice for each burn, once after the coast but before the burn, and once after the burn. In this way, when the trajectory simulation is completed, $2n + 1$ state vectors have been saved, containing all the relevant information about the trajectory.
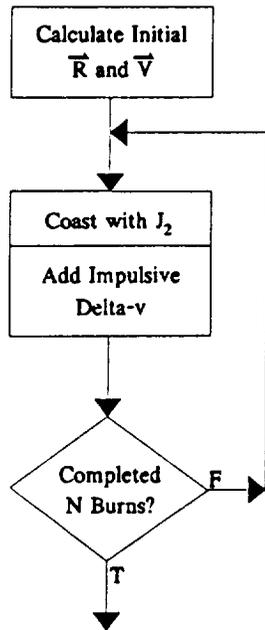

Figure 2 Trajectory Simulation Flow

After completion of the trajectory propagation, the state vector arrays are passed back to the driver interface, where the objective function and constraints are evaluated. If orbital elements are needed for any point in the trajectory, whether at an intermediate point or at the final point, a call to the orbital state calculation routine will translate the ECI state vector into orbital elements corresponding to the desired point in the trajectory. Thus, orbital elements are calculated only when the values are needed by the optimization routines. Typically, constraints are specified to define the final orbit, so orbital elements are calculated for the final state vector. The constraints and objective function may be any smooth, continuous functions dependent upon the independent variables. Constraints may be specified as equalities, inequalities, or a combination of the two, and the objective function may be maximized or minimized.

Use of an analytic state vector propagation routine[7] allows the function generator to simulate a trajectory without the expense of integrating the equations of motion. The analytic propagation routine assumes gravitational acceleration including the J_2 zonal harmonic (i.e., oblate earth), and is capable of propagating a coast in an ellip-

tical or hyperbolic earth orbit. The length of the coast phase typically is input in degrees (argument of the vehicle), although coast duration in seconds also may be used. The J_2 propagation routine is based upon a perturbation expansion of the oblate earth coast, accounting for both short period and secular variations in the orbit. This routine has been in use within GTS for the past several years and has proven to be quite accurate for coasts in earth orbit of up to a few days.

The Driver

The system level driver defines the optimization problem, defines the trajectory to be simulated, and oversees communication among the optimizer, the simulator, and the user. Initialization of the optimization problem includes defining the algorithm control parameters; objective function, independent variable, and constraint scale weights; independent variable upper and lower bounds; and initial values for the independent variables. Initialization of the trajectory simulator includes defining the number of coast-burn pairs, the magnitudes of the delta-v impulses, the initial state vector for the vehicle, and the burn attitudes for the impulsive delta-v's. Burn attitudes may be defined within either an inertial velocity reference frame (VIECI) or an inertial velocity local horizontal reference frame (VIH), and attitudes may be specified using either spherical or cartesian coordinates within the reference frame. When the trajectory to be simulated is defined, any parameters to be used as independent variables for the optimization problem are assigned values from the independent variable array; therefore, it is possible to choose any input parameter as a design variable for optimization.

If an actual upper-stage vehicle is being simulated, subroutines required to define the trajectory sequence for the vehicle are included in the driver portion of the program, tailoring the driver for a specific application. For example, user subroutines can be included to calculate the delta-v magnitude for a given vehicle weight, propellant load, and propellant I_{sp}. This delta-v then can be used by the trajectory simulator in its definition of the trajectory.

During execution, after initializing the optimization and simulation portions of the program, the driver begins by calling the NLP algorithm. A reverse communication technique is used, whereby the NLP algorithm never calls the function evaluator directly. As execution continues, whenever the NLP algorithm requires either a function or gradient computation, control is returned to the driver and the current values for the independent variables are passed in the argument list. The driver then oversees calling of the function generator (i.e., trajectory simulator) or the gradient evaluation routines, and returns the requested data to the NLP algorithm.

The gradient routine[8] used in TROJID calculates two-sided partial derivatives of the constraints and objective function with respect to the independent variables. The routine begins by using user-supplied perturbation sizes for computing the finite-difference derivatives. Error control techniques then are applied to vary the perturbation sizes in order to achieve accurate derivatives. The user has the option to turn off error control, holding perturbation sizes constant throughout execution.

Along with controlling the function generator and gradient routines, the driver also determines what information should be written to the user screen and calls the video screen interface routines to fulfill this task. The video screen interface manages updates to the user screen, providing current data on independent variable, constraint, and objective function values. The interface also tells the user how many function evaluations have been performed, what iteration number is being performed in the search or optimization portions of NLP2, when the postoptimality operator is entered, and when a new Jacobian is being calculated. This information helps the user determine how well the problem is running and whether it should be interrupted and restarted with some sort of modification.

Sample Problems

Several sample problems will be presented in order to show a few of the capabilities available within TROJID. The first problem deals with a generic, two-burn orbit transfer in which the variables are coast durations, delta-v magnitudes, and burn attitudes for the two burns. The objective function to be minimized is the sum of the two delta-v's. The second problem deals with a generic upper-stage vehicle utilizing two solid rocket motors. The objective function to be maximized is the weight of the payload injected into a near geosynchronous mission orbit.

Generic Two-Burn Transfer

Problem 1 is a generic two-burn orbit transfer problem. The objective is to minimize the sum of the two delta-v's required to complete a transfer from an initial orbit to a final orbit. The initial and final orbital elements are defined as

Parameter	Initial Orbit	Final Orbit
apogee altitude	160 nm	25000 nm
perigee altitude	160 nm	300 nm
argument of perigee	*undefined*	90°
inclination	28.5°	67.5°
ascending node	180°	*unconstrained*

The vehicle is initialized at the ascending node. All orbital elements are osculating quantities. The date is initialized as 21 March 1989.

The variables for the problem are

cst_1	coast to first burn (deg)
ΔV_1	first impulsive delta-v (fps)
ψ_1	yaw angle for ΔV_1 (deg)
θ_1	pitch angle for ΔV_1 (deg)
cst_2	coast to second burn (deg)
ΔV_2	second impulsive delta-v (fps)
ψ_2	yaw angle for ΔV_2 (deg)
θ_2	pitch angle for ΔV_2 (deg)

The constraints for the problem are the four orbital parameters defining the final orbit. The objective function is $f = (\Delta V_1 + \Delta V_2)$. The initial guess and solution for this problem are

Variable	Initial Value	Optimal Value
cst_1	90°	94.74970°
ΔV_1	8000 fps	6695.006 fps
ψ_1	0°	2.870789°
θ_1	0°	1.092968°
cst_2	90°	128.0318°
ΔV_2	8000 fps	10767.17 fps
ψ_2	50°	90.12544°
θ_2	0°	27.69700°

The optimal objective function value is 17462.17 fps. The problem was solved in 400 function evaluations and took approximately 48 seconds on a 16 Mhz 80386-based machine with an 80387 numeric coprocessor. Partial derivatives of the optimal objective function with respect to changes in the constraint values are listed in Table 1 along with second partials of the optimal objective function with respect to changes in the variables while feasibility is maintained. A '*' is used to denote the optimal function value. All partials are computed by the postoptimality operator.

Table 1   Sensitivity Results

Parameter	$\partial(\Delta V_1 + \Delta V_2)^*/\partial(\cdot)$
Final Apogee Altitude	0.758616e-2 fps/nm
Final Perigee Altitude	-1.37754 fps/nm
Final Argument of Perigee	125.466 fps/deg
Final Inclination	229.207 fps/deg

	$\partial^2(\Delta V_1 + \Delta V_2)^*/\partial(\cdot)^2$
cst_1	2.05364 fps/deg
ΔV_1	0.274743e-1 fps/fps
ψ_1	1.67737 fps/deg
θ_1	1.34328 fps/deg
cst_2	2.54722 fps/deg
ΔV_2	0.274743e-1 fps/fps
ψ_2	27.1601 fps/deg
θ_2	16.4011 fps/deg

The most sensitive variable is ψ_2, and the least sensitive variables are ΔV_1 and ΔV_2.

The date 21 March was chosen in order to demonstrate the time and sun model capabilities within TROJID. Define the beta angle for an impulsive delta-v to be the angle between the impulsive delta-v vector and the vector from the vehicle to the sun, as shown in Figure 3. A beta angle of 0° implies that the vehicle is pointing directly at the sun during a burn, and a beta angle of 180° implies that the vehicle is pointing directly away from the sun during a burn.
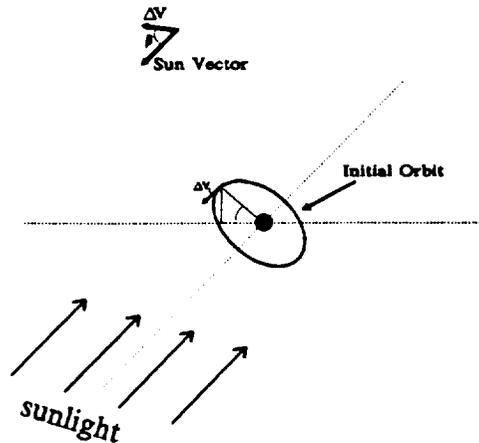

Figure 3 Vehicle-Sun Beta Angle

In the optimal transfer from the example problem above, the beta angles for the first and second burns are 4.9° and 64.1°, respectively. Consider adding a new constraint such that the vehicle is not allowed to point to within 20° of the sun. A new inequality constraint must be added to the problem, requiring that

$$\beta_1 - 20 \geq 0$$

The solution to this new problem is

Variable	Optimal Value
cst_1	101.1227°
ΔV_1	7468.183 fps
ψ_1	18.14728°
θ_1	4.266036°
cst_2	126.7134°
ΔV_2	10173.47 fps
ψ_2	93.88661°
θ_2	25.81913°

The optimal objective function for this new problem is 17641.65 fps, $\beta_1 = 20.0°$, and $\beta_2 = 69.0°$. The increase in the value of β_1 was accomplished through delaying the first burn by about 6.4° of coast arc and increasing the yaw angle for the first burn by about 15.3°. This led to an increase in total delta-v of 179.48 fps. The partial derivative of the minimum total delta-v with respect to changes in the value of β_1 is $\partial f^*/\partial \beta_1 = 24.8506$ fps/deg at the solution to the new problem.

Generic Upper-Stage Transfer

Problem 2 is a generic, upper-stage orbit transfer optimization problem. The objective is to maximize the payload weight injected into the final mission orbit. The upper-stage vehicle consists of two solid rocket motors, SRM1 and SRM2, and a reaction control system (RCS). An RCS trim burn follows each solid rocket motor burn. A propellant margin is set aside in the RCS to account for dispersions. A four-burn simulation is used for the two solid rocket motor and two RCS burns. The first and third burns are the solid rocket motors, with burn attitudes defined in spherical coordinates, while the second and fourth burns are the RCS trim burns, with burn attitudes defined in cartesian coordinates. Some propulsion and weight data are listed in Table 2.

Table 2 Upper-Stage Vehicle Data

SRM1 Propellant Weight	20800 lb
SRM1 I_{sp}	300 sec
Stage 1 Inert Weight	2500 lb
SRM2 Propellant Weight	6300 lb
SRM2 I_{sp}	300 sec
Stage 2 Inert Weight	2000 lb

The initial and final orbital elements are defined as

Parameter	Initial Orbit	Final Orbit
apogee altitude	170 nm	19493 nm
perigee altitude	160 nm	19153 nm
argument of perigee	0°	*unconstrained*
inclination	28.5°	0.5°

The vehicle is initialized at the ascending node. All orbital elements are osculating quantities.

The variables for the problem are

cst_1	coast to first SRM burn (deg)
ψ_1	yaw angle for ΔV_1 (deg)
θ_1	pitch angle for ΔV_1 (deg)
cst_3	coast to second SRM burn (deg)
ψ_3	yaw angle for ΔV_3 (deg)
θ_3	pitch angle for ΔV_3 (deg)
$\Delta V_{2,1}$	x component of ΔV_2 (fps)
$\Delta V_{2,2}$	y component of ΔV_2 (fps)
$\Delta V_{4,1}$	x component of ΔV_4 (fps)
$\Delta V_{4,2}$	y component of ΔV_4 (fps)
pl	payload weight (lb)

The constraints are the three final orbital elements and two constraints that specify the weight of propellant used for the two RCS trim burns to be 55 lb and 65 lb, respectively. The objective function is the payload weight.

The initial guess and solution for this problem are

Variable	Initial Value	Optimal Value
cst_1	0°	-0.3827614°
ψ_1	2°	3.457852°
θ_1	0°	-0.1289974e-1°
cst_2	150°	147.1143°
ψ_3	-50°	-51.21512°
θ_3	0°	0.1007870°
$\Delta V_{2,1}$	20 fps	21.61897 fps
$\Delta V_{2,2}$	0 fps	0.9339027 fps
$\Delta V_{4,1}$	50 fps	50.60754 fps
$\Delta V_{4,2}$	-20 fps	-22.71261 fps
pl	5400 lb	5456.795 lb

The optimal objective function value is 5456.795 lb. The problem was solved in 735 function evaluations and took approximately 121 seconds on a 16 Mhz 80386-based machine with an 80387 numeric coprocessor. Some characteristics of the optimal trajectory design are

SRM1 Delta-v Magnitude	7938.230 fps
SRM2 Delta-v Magnitude	5868.143 fps
Transfer Orbit Apogee Altitude	19325.42 nm
Transfer Orbit Perigee Altitude	160.8560 nm
Transfer Orbit Argument of Perigee	359.7322°
True Anomaly at Final Orbit Injection	291.9432°

Sensitivity results for this problem are listed in Table 3.

Table 3   Sensitivity Results

Parameter	$\frac{\partial pl^*}{\partial(\cdot)}$
SRM1 Propellant Weight	0.4682 lb/lb
SRM2 Propellant Weight	-0.1337 lb/lb
Final Orbit Inclination	48.5807 lb/deg
Final Orbit Apogee	-0.678607 lb/nm
Final Orbit Perigee	-0.686725 lb/nm
Trim Burn 1	1.08061 lb/lb
Trim Burn 2	0.578696 lb/lb

Note that due to the fixed motor sizes of this vehicle, the optimal transfer is not simply a Hohmann transfer with plane change in which injection into the final orbit occurs at apogee (see Figure 4); thus, this is not a problem for which a simple analytically derived solution exists. As can be seen from the partial derivatives, the first motor is undersized and the second motor is oversized for this orbit transfer. Transfer orbit apogee altitude at injection into the transfer orbit is 19325.42 nm, well short of the required final apogee altitude of 19493 nm. If the transfer were reoptimized with final orbit injection constrained to occur at perigee, the optimal payload value would drop to 5452.624 lb, a 4.171 lb loss. If the transfer were reoptimized with final orbit injection constrained to occur at apogee, the optimal payload value would drop to 5420.913 lb, a 35.882 lb loss.
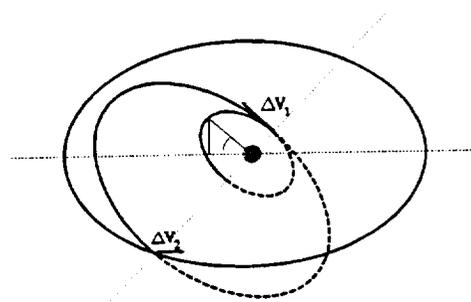

Figure 4 Hohmann Transfer with Plane Change

An alternative problem could be to optimally size the vehicle solid rocket motors in order to minimize total vehicle weight. This problem was run with a constant payload weight of 5400 lb. The optimal motor weights are 20675.84 lb for SRM1 and 6062.297 lb for SRM2. The trajectory produced with these optimal motors is a Hohmann transfer with final orbit injection occurring at apogee.

Future Work

In its current state, TROJID has proven useful for solving both generic, multi-burn orbit transfer problems and vehicle-specific orbit transfer problems. In the near future, the ability to integrate constant thrust and Isp burns will be added, allowing a higher degree of fidelity when specific vehicles with known motor characteristics are simulated. Since the burns currently are being modeled with an instantaneous delta-v impulse, integration of the burns will increase the amount of computation time required for each function evaluation (trajectory simulation), although this increased computation time should not be excessive for burn times of relatively short duration (on the order of hundreds of seconds). A natural use for this added capability will be first to optimize an impulsive burn problem; then, using this approximate solution as an initial guess, to reoptimize using integrated burns.

Conclusion

This paper has described TROJID, a trajectory optimization package capable of producing optimal trajectory designs for multi-burn, upper-stage orbit transfers. The package utilizes a state-of-the-art nonlinear programming algorithm along with a trajectory simulator capable of simulating impulsive delta-v trajectories with an analytic coast phase propagator incorporating the J_2 zonal harmonic. The package has been demonstrated to solve both generic, multi-burn orbit transfer trajectory optimization problems and upper-stage trajectory optimization problems in which a specific upper-stage vehicle is modeled in the simulation.

References

1 "The Generalized Trajectory Simulation System," Volumes 1-5, Report No. TR-0075(5549)-1, The Aerospace Corporation, El Segundo, CA, 30 June 1975.

2 Brauer, G.L., et al., "Program to Optimize Simulated Trajectories (POST)," Volumes 1-3, Report No. MCR-87-583, NAS1-18147, Martin Marietta Corporation, Denver, CO, September 1987.

3 Betts, J. T. and W. P. Hallman, "NLP2 Optimization Algorithm Documentation," Report No. TOR-0089(4464-06)-1, The Aerospace Corporation, El Segundo, CA, 28 August 1989.

4 Beltracchi, T.J., "An Overview of the Current State of the Art Optimization Used for Trajectory Design in the GTS System," Submitted to the Third Annual Air Force/NASA Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, San Francisco, CA, September 1990.

5 Hallman, W.P., "A Software Package for Postoptimality Analysis of the Nonlinear Programming Problem," Report No. ATM-86(6464-06)-5, The Aerospace Corporation, 29 April 1986.*

6 Hallman, W.P., "Sensitivity Analysis for Trajectory Optimization Problems," AIAA Paper 90-0471, AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, January 1990.

7 Huffman, W.P., "An Analytic J_2 Propagation Metho ' for Low Earth Orbits," Internal Document, ' Aerospace Corporation, El Segundo, CA, No 1981.*

8 Hallman, W.P., "Numerical Derivative ' Trajectory Optimization," Submitt Annual Air Force/NASA Sympo vances in Multidisciplinary A tion, San Francisco, CA, S'

---

* Aerospace internal report; not available for external distribution

ON DOMAINS OF CONVERGENCE
IN OPTIMIZATION PROBLEMS

Alejandro R. Díaz, Steven S. Shaw, and Jian Pan
Department of Mechanical Engineering
Michigan State University
East Lansing, MI 48824-1226

INTRODUCTION

Numerical optimization algorithms require the knowledge of an initial set of design variables. Starting from an initial design x^0, improved solutions are obtained by updating the design iteratively in a way prescribed by the particular algorithm used. If the algorithm is successful, convergence is achieved to a *local* optimal solution. Let A denote the iterative procedure that characterizes a typical optimization algorithm, applied to the problem:

> Find $x \in R^n$ that
> > maximizes $f(x)$
> > subject to $x \in \Omega \subseteq R^n$

We are interested in problems with several local maxima x_j^*, j=1, ..., m, in the feasible design space Ω.

In general, convergence of the algorithm A to a *specific* solution x_j^* is determined by the choice of initial design x^0. The domain of convergence Dj of A associated with a local maximum x_j^* is a subset of initial designs x^0 in Ω such that the sequence $\{x^k\}$, k=0,1,2,... defined by

$$x^{k+1} = A(x^k), \quad k=0,1,...$$

converges to x_j^* , that is,

$$D_j = \{x \in \Omega : \ \|x_j^* - x^k\| \to 0 \ \text{as} \ k \to \infty \ \text{whenever} \ x^0 = x \ \text{and} \ x^{k+1} = A(x^k) \}$$

The set Dj is also called the *basin of attraction* of x_j^*.

Cayley [1] first proposed the problem of finding the basin of attraction for Newton's method in 1897. It has been shown that the basin of attraction for Newton's method exhibits chaotic behavior in problems with polynomial objective [2, 3]. This implies that there may be regions in the feasible design space where arbitrarily close starting points will converge to different local optimal solutions. Furthermore, the boundaries of the domains of convergence may have a very complex, even fractal structure. In this paper we show that even simple structural optimization problems solved using standard gradient based (first order) algorithms exhibit similar features.

STRUCTURE OF DOMAINS OF CONVERGENCE

The following example is used to illustrate the more relevant features of the problem. Consider the beam shown in Figure 1. Translation is constrained by rigid supports at both ends and rotation is restricted through rotational springs of finite stiffness r. A finite element model is used for the analysis discretizing the beam into four tapered beam elements. The element heights b_1, b_2 and b_3 at nodal points are the chosen design variables. The optimization problem considers the maximization of the smallest eigenvalue of the beam, λ_1, under an isoperimetric constraint on the beam volume, i.e., the problem (1) below:

Find $\{b_1, b_2, b_3\}$ that
maximize $\quad \lambda_1$
subject to $\quad b_1 + b_3 + 2b_2 = C$
$\qquad\qquad b_i \geq 0$ $\hspace{4cm}$ (1)

For the specific case considered here, we let $C=4$ and $r=1.0$, a relative large value corresponding to nearly clamped ends. The objective function has three local maxima and two saddle points.
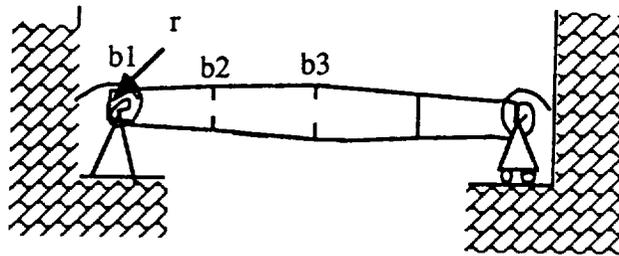

Figure 1. The beam model

A first-order, gradient based optimization algorithm is used to solve problem (1). To facilitate the graphical representation of the domains of convergence, the equality constraint problem is used to restate the problem as follows:

Find $x = \{b_1, b_3\}$ that
maximize $\quad \lambda_1$
subject to $\quad b_1 + b_3 \leq 4$
$\qquad\qquad b_1 \geq 0, \quad b_3 \geq 0$ $\hspace{3cm}$ (2)

The feasible set is the two-dimensional region

$$\Omega = \{\{b_1, b_3\} \in R^2 : b_1 + b_3 \leq 4, \ b_1 \geq 0, b_3 \geq 0\}$$

The algorithm used is a steepest ascent algorithm modified to account for the simple constraints. It can be characterized by the mapping

$$A(x^k) = x^k + \alpha(x^k) s(x^k)$$ $\hspace{3cm}$ (3a)

where
$$s(x^k) = \nabla f(x^k) \qquad \text{if } x^k \in \text{Int}(\Omega)$$ $\hspace{2cm}$ (3b)
$$s(x^k) = P_{\partial\Omega}[\nabla f(x^k)] \qquad \text{if } x^k \in \partial\Omega$$ $\hspace{2cm}$ (3c)

$\partial\Omega$ is used to denote the boundary of Ω and $P_{\partial\Omega}[\nabla f(x^k)]$ denotes a projection of $\nabla f(x^k)$ onto $\partial\Omega$. The scalar parameter α is the step size obtained from line search along the direction s. Both exact and inaccurate line search based on Armijo's rule are used here.

For exact line search:

$$\alpha(x^k) = \max \{ \alpha \geq 0 : f(x^k + \alpha s(x^k)) \rightarrow \max \} \tag{4}$$

For inaccurate line search based on Armijo's rule:

$$\alpha(x^k) = \max_{i=0,1,\ldots} \{ \alpha_i : f(x^k + \alpha_i s(x^k)) - f(x^k) - \delta \alpha_i |\nabla f(x^k)| \geq 0 , \alpha_i = \varepsilon^i \alpha_{max} \} \tag{5}$$

for fixed values of δ and ε (0.02, 0.5, resp). $\alpha_{max} > 0$ is such that $x^k + \alpha_{max} s(x^k)$ is on $\delta\Omega$. The difference between these two strategies is illustrated in Figures 2 and 3.
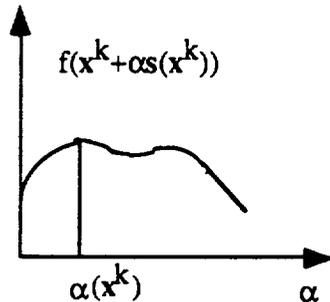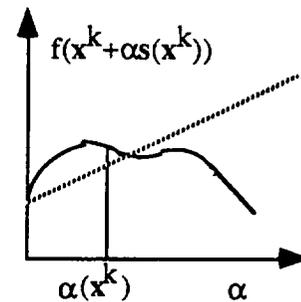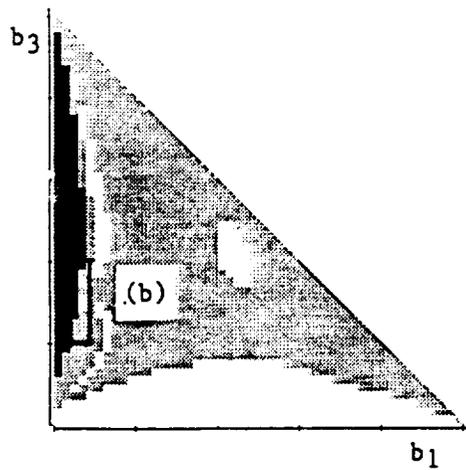

Figure 2 Exact line search

Figure 3 Inaccurate line search based on Armijo's rule

To study the structure of the domains of convergence, the feasible design space is divided into a two-dimensional grid of equally spaced, discrete starting points x^0_i. The algorithm A is applied starting from each grid point until convergence to one of the 3 maxima, labeled x_1^*, x_2^*, x_3^*, is achieved (Notice that for the values of the parameters used eigenvalue crossing, a well known phenomenon that is common in eigenvalue maximization problems, *does not occur*). A small rectangular region R_i centered around x^0_i is assumed to be in the domain of convergence D_j of x_j^* if the distance between $A(x^k)$ and x_j^* is smaller than a prescribed tolerance for some K and $k \geq K$. R_i is colored according to the following scheme:

dark   if   A converged to x_1^* from x^0_i

gray   if   A converged to x_2^* from x^0_i

light   if   A converged to x_3^* from x^0_i

The resulting figures are shown in Figure 4 for exact line search and in Figure 5 for inaccurate line search. It is clear from the figures that the domains of convergence are structurally different for the two line search methods. In the case of inaccurate line search, the domains of convergence have an extremely fine layered structure in some regions of Ω that is not present in this problem when exact line search is used. The structural difference in the domains of convergence of the two methods is dominated by the difference between the function $\alpha(x)$ corresponding to exact and inaccurate line search. These results indicate that commonly used optimization algorithms based on first order information have features similar to those found in Newton's method: sensitive dependence on initial design and very fine layered domains of convergence. The mappings which result from these algorithms cannot be chaotic due to their inherent ascent properties, but they can lead to very complicated, possibly fractal, boundaries separating domains of convergence. We conjecture that the source of this complexity is the same as that which leads to similar boundaries in other dynamical systems, although the connection is not clear due to the discontinuous nature of the optimization mappings, particularly those associated with inaccurate line search strategies. To describe this we need some concepts and definitions borrowed from the qualitative theory of dynamical systems.

200
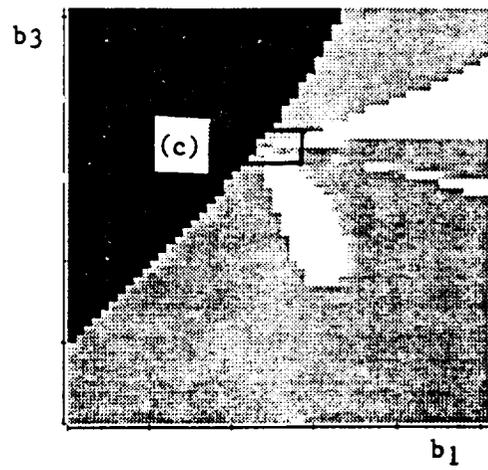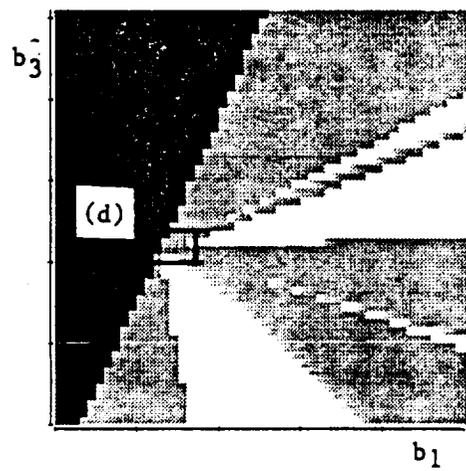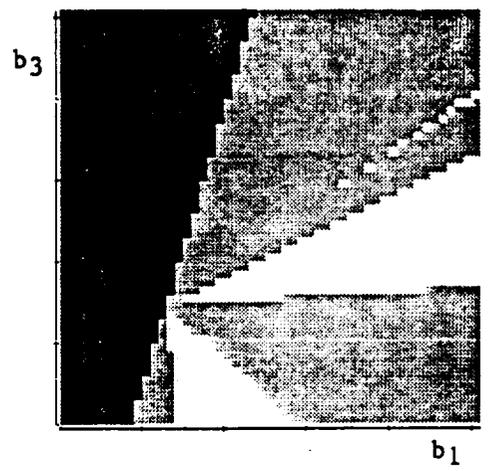
Figure 4 Domains of convergence for exact line search

(a)   [0.0, 4.0] X [0.0, 4.0]

(b)   [0.196, 0.342] X [0.83, 1.61]

(c)   [0.25732, 0.27776] X [1.3292, 1.3926]

(d)   [0.2618168, 0.2642696] X [1.35456, 1.359632]
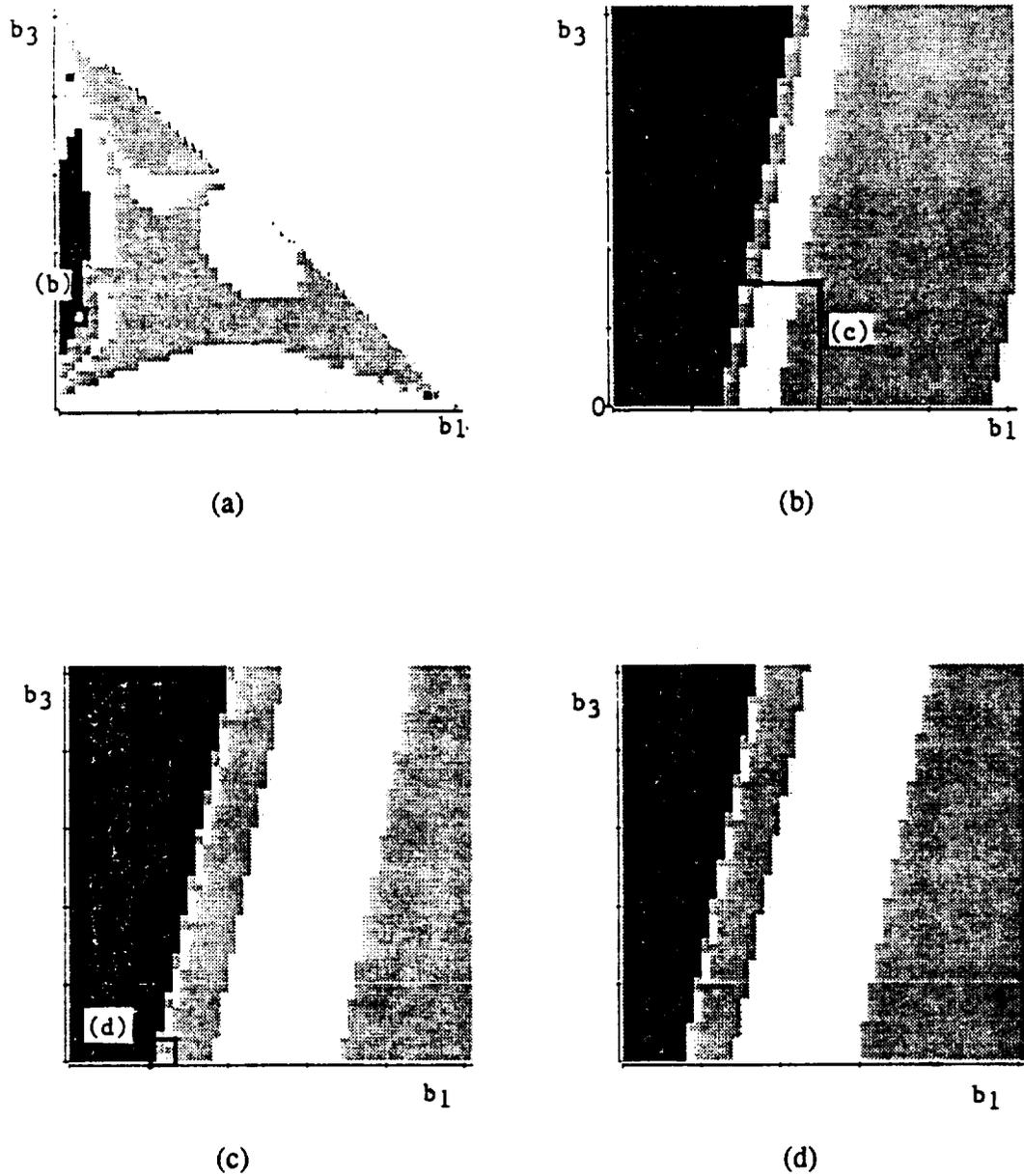
Figure 5   Domains of convergence for inexact line search

(a)   [0.0, 4.0] X [0.0, 4.0]

(b)   [0.18, 0.20] X [0.95, 0.97]

(c)   [0.184, 0.190] X [0.950, 0.956]

(d)   [0.1852, 0.18556] X [0.950, 0.95036]

First, note that all local maxima, local minima, and saddle points are fixed points of these maps. The set of points which are mapped to, or are asymptotic to, a fixed point is defined as the stable manifold of that fixed point (for example, the stable manifold of a local maximum is its domain of convergence). Similarly, the unstable manifold of a fixed point is the set of points which are mapped to, or are asymptotic to, the fixed point under inverse iterations of the map (when the map is not strictly invertible, one can use the idea of pre-image points). Note that the stable manifold of a saddle point is the boundary between points which branch away from the saddle in different directions (it is often referred to as a separatrix). For continuous maps, the source of complexity is the intersection of the stable manifold of one saddle point with the unstable manifold of another saddle. Such an intersection leads to the stable manifold, and therefore the boundary, winding in a very complicated manner which results in the infinitely fine layered structure for domains of convergence, and it the most likely cause of the observed behavior.

We note that in order for such intersections to occur at least two saddle points must exist, which implies that at least three local maxima must exist. In a system with only two local maxima the boundary separating their domains of convergence cannot, if our conjecture is correct, have this layered quality. The problem solved here has three local maxima. However, if the stiffness of the rotational springs r is reduced to a sufficiently low value, the number of maxima changes. In the case r=0, corresponding to a pined-pined beam, the objective function has only one maximum and hence the domain of convergence is Ω. This observation indicates that the chances of success of the overall solution strategy can be affected significantly by possibly small changes in the models used to represent the *physical* boundary conditions. Extensions to problems involving more than two design variables can be made, but one must be careful since the domain boundaries are formed only by the stable manifolds of saddles for which the dimension of the stable manifold is one less than the design space.

REFERENCES

1. Cayley, A.: Application of the Newton-Fourier Method to An Imaginary Root of An Equation, Quart. J. Pure Appl. Math. 16, 1879, pp. 179-185.

2. Peitgen, H.O., Prufer, M., Schimitt K.: Global Aspects of the Continuous and Discrete Newton Method: A Case Study. Acta Appl. Math., 13, 1988, pp. 123-202.

3. Benzinger, H.E., Burns, S.A., Palmore, J.I.: Chaotic Complex Dynamics and Newton's Method, Physics Letters A, Vol. 119, No. 9, 19 Jan. 1987, pp. 441-446.

A SUPERLINEAR INTERIOR POINTS ALGORITHM FOR ENGINEERING
DESIGN OPTIMIZATION

J. Herskovits *
J. Asquier **

ABSTRACT:

We present a quasi-Newton interior points algorithm for nonlinear constrained optimization. It is based on a general approach consisting on the iterative solution in the primal and dual spaces of the equalities in Karush-Kuhn-Tucker Optimality Conditions. This is done in such a way to have primal and dual feasibility at each iteration, which ensures satisfaction of those optimality conditions at the limit points.

This approach is very strong and efficient, since at each iteration it only requires the solution of two linear systems with the same matrix, instead of Quadratic Programming Subproblems. It is also particularly appropriated for Engineering Design Optimization inasmuch at each iteration a feasible design is obtained.

The present algorithm uses a Quasi-Newton approximation of the second derivative of the Lagrangian Function in order to have superlinear asymptotic convergence. We discuss about theoretical aspects of the algorithm and it's computer implementation.

1. INTRODUCTION

In this work we present a quasi-Newton interior points algorithm for nonlinear inequality constrained optimization, based on a general approach which solves Karush-Kuhn-Tucker optimality criteria by means of fixed point iterates in the primal and dual space [4]. Given an initial point in the interior of the feasible region, a sequence of interior points is generated in such a way that the objective decreases monotonically and which converges to a K-K-T point of the problem.

This approach is particularly appropriated for Engineering Design Optimization since at each iteration a feasible design is obtained and it is also very strong and efficient.

In the next section, we consider the inequality constrained problem and discuss the basic ideas of the method. A quasi - Newton algorithm is presented in the following section and then a technique for updating the approximation matrix. Finally we briefly discuss some aspects about the computer implementation of the algorithm.

---

*   COPPE/Federal University of Rio de Janeiro, Mechanical Engineering Dept, Caixa Postal 68503, 21945 Rio de Janeiro, Brazil.

** AÉROSPATIALE - Cannes, 100 BD du Midi, Cannes - La Bocca, 06400 and Visiting Research Fellow at COPPE/Federal University of Rio de Janeiro.

2. THE INEQUALITY CONSTRAINED PROBLEM

We consider now the inequality constrained nonlinear programming problem :

$$\text{minimize } f(x)$$
$$\text{submitted to } g(x) \le 0, \qquad (1)$$

where $f \in R$ and $g \in R^m$ are smooth functions in R^n. This problem is normally present in Engineering Design, however equality constraints can easily be included using this approach [1].

Definitions

Definition 1. $d \in R^n$ is a *descent direction of f at x if $d^t \nabla f < 0$. $\quad \square$

Definition 2. $d \in R^n$ is a *feasible direction* of the problem at $x \in \Omega$, $\Omega = \{x \in R^n / g(x) \le 0\}$, if for some $\theta > 0$ we have $x + td \in \Omega$ for all $t \in (0,\theta)$. $\quad \square$

Definition 3. A vector field $d(x)$ defined on $\Omega \in R^n$ is said to be a *uniformly feasible directions field* if there exists $\tau > 0$ such that for any $x \in \Omega$ $x + td(x) \in \Omega$ for all $t \in (0,\tau)$. $\quad \square$

The present algorithm obtains at each iteration a search direction d, which is a descent direction of the objective and also a feasible direction of the problem. A line search is then performed to ensure that the new point is interior and the objective is lower. As a consequence of the requirement of feasibility, d must actually constitute an uniformly feasible directions field. Otherwise, the step length may go to zero and convergence to non-stationary points may occur.

Karush-Kuhn-Tucker first order optimality conditions consist on the following system of equalities and inequalities:

$$C + A^t \lambda = 0, \qquad (2)$$

$$G \lambda = 0, \qquad (3)$$

$$g(x) \le 0 \text{ and} \qquad (4)$$

$$\lambda \ge 0, \qquad (5)$$

where $\lambda \in R^m$ is the dual variables vector, $C = \nabla f(x)$, $A = \nabla g(x)$, and $G = \text{diag } [g(x)]$, $G \in R^{mxm}$.

Our approach consists on solving the system of equations (2),(3) in (x,λ) by means of a quasi-Newton algorithm. This is done in such a way to satisfy inequalities (4) and (5) at each iteration, in order to ensure that Karush-Kuhn-Tucker conditions are verified at the limit points.

Consider the fixed point iterates for the solution of (2),(3) defined by the following linear system of equations:

$$\begin{bmatrix} B & A^t \\ \Lambda A & G \end{bmatrix} \begin{bmatrix} x_0 - x \\ \lambda_0 - \lambda \end{bmatrix} = - \begin{bmatrix} C + A^t \lambda \\ G\lambda \end{bmatrix} \tag{6}$$

where $B \in R^{nxn}$ is symmetric and positive definite, $\Lambda \in R^{mxm}$ is $\Lambda = diag(\lambda)$, (x,λ) is the actual iterate and (x_0,λ_0) is a new estimate. Taking $B = H(x,\lambda)$, where $H(x,\lambda) = \nabla^2 f(x) + \sum \lambda_i \nabla^2 g_i(x)$, (6) becomes the Newton – Raphson's iterates. In the present paper, we take B equal to a quasi-Newton approximation of $H(x,\lambda)$.

Let be $d_0 = x_0 - x$. Then, (6) becomes

$$Bd_0 + A^t \lambda_0 = - C \quad \text{and} \tag{7}$$

$$\Lambda A d_0 + G\lambda_0 = 0, \tag{8}$$

which now gives a direction d_0 in the primal space. It can be proved, in a similar way as in [1,3], that d_0 is a descent direction of f. However, d_0 is not useful as a search direction since it does not always constitute an uniformly feasible directions field. This is due to the fact that as any constraint goes to zero, (8) forces d_0 to tend to a direction tangent to the feasible set.

This effect is avoided by including a negative vector $(-\mu e)$ in the right side of (8), where the scalar factor μ is positive and $e = [1,1,...,1]^t$, $e \in R^m$. Then, solving

$$Bd + A^t \bar{\lambda} = - C, \tag{9}$$

$$\Lambda A d + G\bar{\lambda} = - \mu e \tag{10}$$

we have a new direction d which constitutes an uniformly feasible directions field [1].

Finally, since d can be considered as a perturbation proportional to μ of the descent direction d_0, it is possible to establish bounds on μ which ensure that d is also a descent direction of f.

The ideas pointed above are a basis for the iterative method that we are studying. In the primal space, a line search is done in direction of d while, in the dual space, updating of λ is defined by

$$\lambda_i := sup[\lambda_{0i} ; \epsilon \|d_0\|^2]; \quad i = 1,m, \quad \epsilon > 0, \tag{11}$$

which ensures that λ is always feasible.

3. THE ALGORITHM

The algorithm that we present is stated as follows:

Parameters. $\alpha \in (0,1)$, $\gamma \in (0,1)$, $\varepsilon > 0$, η_1 and $\eta_2 \in (0,1)$, $(\eta_1 > \eta_2)$, $\bar{\mu} > 0$ and $\nu \in (0,1)$.

Data. $x \in \Omega$, $\lambda > 0$ and $B \in R^{n \times n}$ symmetric and positive definite.

Step 1. Computation of a search direction.

(i)   Compute (d_0, λ_0) by solving the linear system

$$B\, d_0 + A^t \lambda_0 = -C, \tag{12}$$

$$\Lambda A\, d_0 + G\, \lambda_0 = 0. \tag{13}$$

   If $d_0 = 0$, stop.

(ii)   Compute (d_1, λ_1) by solving the linear system

$$B d_1 + A^t \lambda_1 = 0, \tag{14}$$

$$\Lambda A d_1 + G \lambda_1 = -e. \tag{15}$$

(iii)   Compute the search direction

$$d = d_0 + \mu^* d_1, \tag{16}$$

   where $\mu = \sup \{\bar{\mu}; \ \sup\ [(\alpha-1) d_0^t C / d_1^t C;\ 0]\}$. $\tag{17}$

Step 2. Line search.

   Find $t > 0$ such that:

   $f(x + td) \le f(x) + t\, \eta_1 d^t C$, and

   $g_i(x + td) < 0$; $i = 1, m$,

   are true, and at least one of the following conditions is also true:

   $d^t \nabla f(x+td) \ge \eta_2 d^t C$, or

   $g_i(x+td) \ge \gamma\, g_i(x)$; $i = 1, 2, \ldots, m$.

Step 3. Updates.

(i)   Set

   $x := x + td$, and

   $\lambda_i := \sup[\lambda_{0i}; \varepsilon \|d_0\|^2]$; $i = 1, m$.

(ii)   Compute a new symmetric positive definite approximation

   B to $H(x, \lambda)$.

(iii)   Go back to *Step 1.*   □

The search direction d given by (16) is the same as the one obtained in (10) and condition (17) on μ ensures that

$$d^t C \le \alpha\, d_0^t C$$

207

Then, since d_0 is a descent direction of f, d also is.

The line search criteria is very wide and easy to satisfy. We find the step length in an iterative way by defining an initial t and, if the criteria is not verified, making quadratic interpolations or extrapolations until a satisfactory step length is obtained.

4. THE QUASI - NEWTON APPROXIMATION MATRIX

In unconstrained optimization problems by quasi - Newton method, an approximation matrix B to the second derivative of the function is built up. The formula preferred by several authors for updating B is the BFGS rule

$$B := B - \frac{B\delta\delta^t B}{\delta^t B\delta} + \frac{\eta\eta^t}{\delta^t\eta} \tag{18}$$

where $\delta \in R^n$ is the change of the variables and $\eta \in R^n$ is the change of the gradient of the function. Since the second derivative of the function is positive definite at the minimum, it can be proved that if the initial B is positive definite a new approximation matrix with the same property is obtained, provided that

$$\delta^t\eta > 0. \tag{19}$$

In the present algorithm, B is an approximation to $H(x,\lambda)$ which is not necessarily positive definite at the solution. We let $\gamma \in R^n$ to be the change in x of $\{\nabla f(x) + A^t(x)\lambda\}$ and, in order to satisfy (19), we adopt a procedure proposed by Powell [5] for the definition of η.

Consider $\phi \in R$ defined as follows:

$\phi = 1,$ if $\delta^t\gamma \geq 0.2 \, \delta^t B\delta$, or

$\phi = \dfrac{0.8 \, \delta^t B\delta}{\delta^t B\delta - \delta^t\gamma}$ otherwise.

Then, we define η to be

$\eta = \phi\gamma + (1-\phi)B\delta$

and B given by the updating rule (18).

5. NUMERICAL IMPLEMENTATION

Several items involved in the implementation of the present algorithm merit a wider discussion. One question is that of efficiently solving linear systems (12,13) and (14,15). It can be proved that the corresponding matrix is nonsingular [3]; however it is not symmetric neither positive definite. Different technics can be employed which lead to systems involving symmetric

and positive definite matrices with smaller dimensions. Another important point is that of the line search procedure, since a good implementation can result in important benefits in the global efficiency of the algorithm. Quasi - Newton updating can be done as in (18) but it is also possible to approximate the inverse of $H(x,\lambda)$ or to generate Cholesky's decomposition of both approximation matrices.

REFERENCES:

[1] Herskovits, J. "A two-stage feasible directions algorithm for nonlinear constrained optimization", Mathematical Programming, Vol. 36, pp. 19-38, 1986.

[2] Herskovits, J. "A two-stage feasible directions algorithm including variable metric techniques for nonlinear optimization", Research Report n· 118, INRIA, BP 105, 78153 Le Chesnay CEDEX, France, 1982.

[3] Panier, E.R.,Tits A.L. and Herskovits J. "A QP - Free, globally convergent, locally superlinearly convergent algorithm for inequality constrained optimization", SIAM Journal of Control and Optimization, Vol 26, pp 788-810, 1988.

[4] Herskovits, J. and Coelho, C.A.B. "An interior points algorithm for structural optimization problems, Proceedings of the first International Conference", pp. 231-241, Southampton, UK, June 1989.

[5] Powell, M.J.D. "The convergence of variable metric methods for nonlinearly constrained optimization calculations", in Numerical Methods for Constrained Optimization, edited by P.E.Gill and W.Murray, Academic Press, London, 1974.

SHAPE DESIGN SENSITIVITIES USING FULLY AUTOMATIC 3-D MESH GENERATION

M. E. Botkin
General Motors Research Laboratories
Warren, Michigan 48090-9057

ABSTRACT

Previous work in 3-D shape optimization involved specifying design variables by associating parameters directly with mesh points. More recent work has shown the use of fully-automatic mesh generation based upon a parameterized geometric representation. Design variables have been associated with a mathematical model of the part rather than the discretized representation. The mesh generation procedure uses a nonuniform grid intersection technique to place nodal points directly on the surface geometry. Although there exists an associativity between the mesh and the geometrical/topological entities, there is no mathematical functional relationship. This poses a problem during certain steps in the optimization process in which geometry modification is required. For the large geometrical changes which occur at the beginning of each optimization step, a completely new mesh is created. However, for gradient calculations many small changes must be made and it would be too costly to regenerate the mesh for each design variable perturbation. For that reason, a local remeshing procedure has been implemented which operates only on the specific edges and faces associated with the design variable being perturbed. Two realistic design problems are presented which show the efficiency of this process and test the accuracy of the gradient computations.

SHAPE PARAMETERIZATION

Of the many aspects of a shape optimization system, the most controversial issue, and the issue still unresolved, is the process by which design variables are assigned to the mesh. There are essentially four approaches being investigated, two dealing with some form of mesh generation and two which work from an existing mesh. The method based upon mapped mesh generation[1,2] requires the association of design parameters with key nodes of 3-D iso-parametric hyper-patches. In this case, as the shape changes, internal nodes are moved automatically. The method developed by Yang[3] assumes the existence of a mesh generated manually using a conventional modeling system. Using graphics, design variables can be directly assigned to mesh points. No attempt is made to move internal points. A third method, originally developed by Belegundu and Rajan[4,5] also assumes a mesh exists. This approach however uses fictitious loads as design variables. The deformed shapes resulting from the application of special loading and boundary condition sets are used as design variations. The advantage of this method is that interior node movements are directly obtained. All of the above techniques pose certain problems in developing the design models. For that reason Botkin[6] proposed the use of fully automatic mesh generation based upon the use of a parameterized 3-D mathematical surface model. For each step in the optimization process, the current set of design variables produces a new geometrical model from which a mesh can be automatically generated. There is no need in this case to relate finite element nodes to design variables and internal nodes are moved automatically. Since the first three methods directly assign nodes to design variables, shape sensitivities are easily obtained, i.e., small perturbations in design variables can produce small perturbations in mesh points. However, no such direct

relationship exists for the fourth approach and special consideration will be given to this aspect of the problem in the next section.

SENSITIVITY ANALYSIS

The following relationship expresses the association between the geometric model, G and the mesh, M:

$$G(\delta) \rightarrow M \tag{1}$$

in which δ represents the design variables. Whereas the model is an explicit function of the design variables, the mesh is only obtained through the operation $\rightarrow$.

Using the symbolic relationship (1) the behavioral sensitivities can be computed as follows:

$$\frac{\partial R}{\delta \delta} = \frac{\partial R}{\delta M} \times \frac{\partial M}{\delta G} \times \frac{\partial G}{\delta \delta} \tag{2}$$

in which R is a response quantity. Most of the questions regarding shape sensitivities have been answered[7] but have dealt only with the first term of Eq. 2. Term three is analytical and can be easily computed. One aspect as it pertains to automatic mesh generation still has to be investigated, and deals with term two. As the design variables are sequentially perturbed, a resulting perturbed mesh must be obtained. Figure 1 represents a two-dimensional segment of a mesh, the associated geometry, and design variables using the notation of relationship (1). The bold curve represents a typical shape variation. The subscripted variable M_o refers to mesh points on the surface geometry and variable M_i refers to points in the interior. It would not be desirable to regenerate an entirely new mesh for each design variable perturbation, $\Delta\delta_i$, but only recompute the surface point locations, M_o, locally. Since the mesh is tied to the geometrical description through an operation, it is only indirectly associated with the design variables. For that reason a local remeshing procedure was implemented which operates only upon the mesh data for specific edges and faces. Associativity information between the topology and the mesh allows only pertinent nodes to be identified for repositioning. There is still some question about the accuracy of this localized numerical process when only small changes are required as in the sensitivity calculations. In an attempt to improve the accuracy a methodology was developed in which rather substantial boundary movements are first performed to obtain the direction of the velocity field[7] and then scaled to the small changes required. Finally, it still may be necessary to devise an approach, such as Laplacian smoothing, to move internal points, M_i, as well to use as velocities in Ref. 7. This aspect of the problem should be investigated further since there is a possibility that too much nonlinearity exists in such a smoothing operation to produce accurate sensitivities.

DESIGN EXAMPLES

The following examples were modeled using the mesh generation capability developed by Shephard and Yerry[8,9] and documented in Reference 6. Computer times given are for an IBM 3090. Response sensitivities were computed using a special version of NASTRAN in which semi-analytic grid sensitivities were implemented.

Two examples are used to investigate the efficiency and accuracy of shape sensitivities using automatic mesh generation. For an idealized torque arm shown in Fig. 2 and an upper control arm shown in Fig. 3 the model size information is given in Table 1. The torque arm is loaded at one hole and clamped at the other. The control arm is loaded at the small end and clamped at both holes on the large end. The geometrical models are shown in Figs. 4 and 5. The cross-hatched surfaces in these figures indicate the design variation used to compute sensitivities. In an actual design study, it would be desirable to allow all surfaces to vary, but for this example only a single surface will be considered. Behavioral sensitivities will be computed for the displacement at the loaded node with respect to the specific surface movement taking the form of a quadratic function. Stress sensitivities were not computed since they are merely a function of displacement sensitivities. Table 2 gives the perturbation data. Only a small fraction of the initial mesh generation time is required to locally update the mesh for sensitivity calculations and the sensitivities agree quite well with the finite difference values. It is believed however that the analytical values are more accurate than the finite difference values. It is very difficult to accurately input the perturbed mesh data to NASTRAN, even using extended precision fields, whereas the grid point sensitivity data for the analytical calculation requires the velocity[7] values which can be input much more accurately.

Table 1 Mesh Generation Data

	NODES	SOLIDS	MESH GENERATION TIME
TORQUE ARM	826	2875	35 sec
CONTROL ARM	975	3192	49 sec

Table 2 Mesh Sensitivity Data

	MESH PERTURBATION TIME	SENSITIVITY	F.D. SENSITIVITY
TORQUE ARM	1.3 sec	.00280	.00286 (2%)
CONTROL ARM	1.8 sec	.00405	.00401 (-1%)

REFERENCES

1. M. E. Botkin, "Shape Optimization of Components: Plate and Shell Applications," AIAA Journal, Vol. 20, No. 2, pp. 268-273(1982).
2. M. H. Imam, "Application of Shape Optimization to the Engine Main Bearing Cap," ASME Computers in Engineering, Vol. 3, pp. 119-126(1982).
3. R. J. Yang, "SHOP-3D: Shape Optimization of 3-D Structural Components," Computers & Structures, Vol. 31, No. 6, pp. 881-890(1989).
4. A. D. Belegundu and S. D. Rajan,"A Shape Optimization Approach Based Upon Natural Design Variables and Shape Functions," Computer Methods in Applied Mechanics and Engineering, Vol. 36, pp 87-106, 1988.
5. S. D. Rajan and A. D. Belegundu,"Shape Optimal Design Using Fictitious Loads," AIAA Journal, Vol. 27(1), pp 102-107, 1989.
6. M. E. Botkin,"Shape Design Modeling Using Fully-Automatic 3-D Mesh Generation," AIAA SDM Conference, AIAA-90-1009, pp. 171-183(1990).
7. R. J. Yang and M. E. Botkin,"Accuracy of the Domain Method for the Material Derivative Approach to Shape Design Sensitivities," AIAA Journal, Vol. 25, No. 12, pp. 1606-1610(1987).
8. M. A. Yerry and M. S. Shephard,"Automatic Three-Dimensional Mesh Generation by the Modified Octree Approach," International Journal of Numerical Methods in Engineering, Vol. 20, pp 1965-1990, 1984.
9. M. A. Yerry and M. S. Shephard,"Automatic Three-Dimensional Mesh Generation for Three-Dimensional Solids," Computers in Structures, Vol. 20, pp 31-39, 1985.
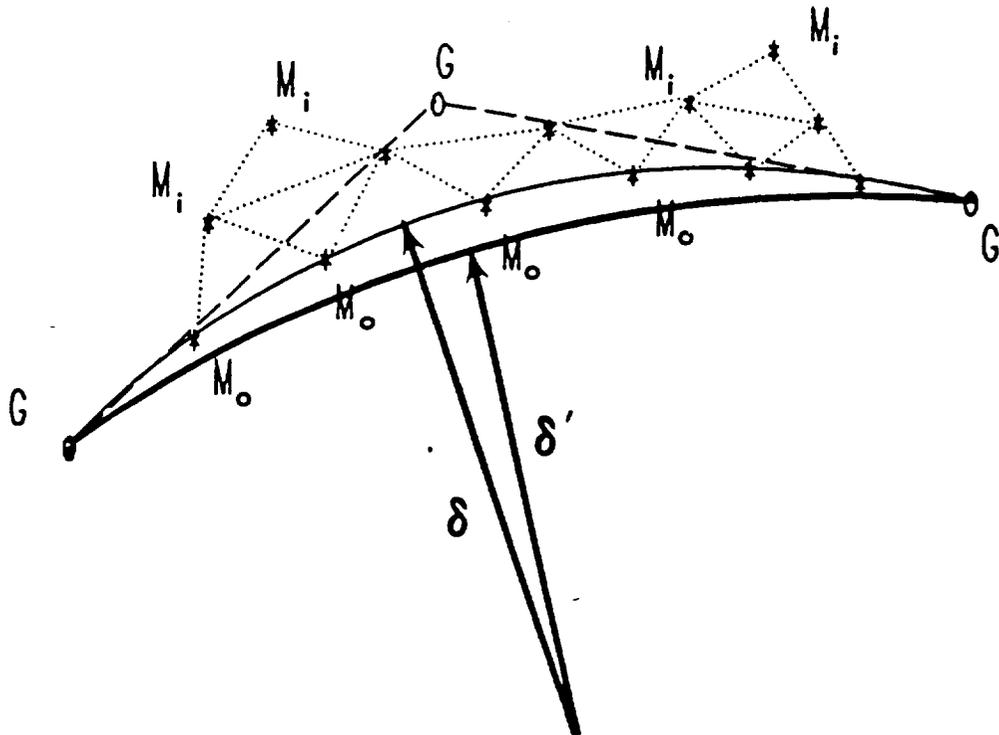
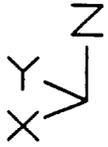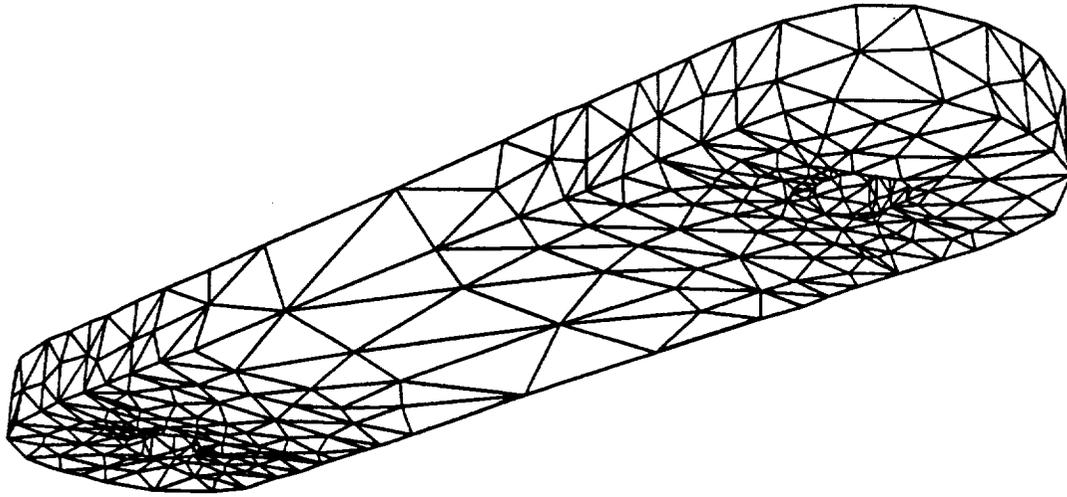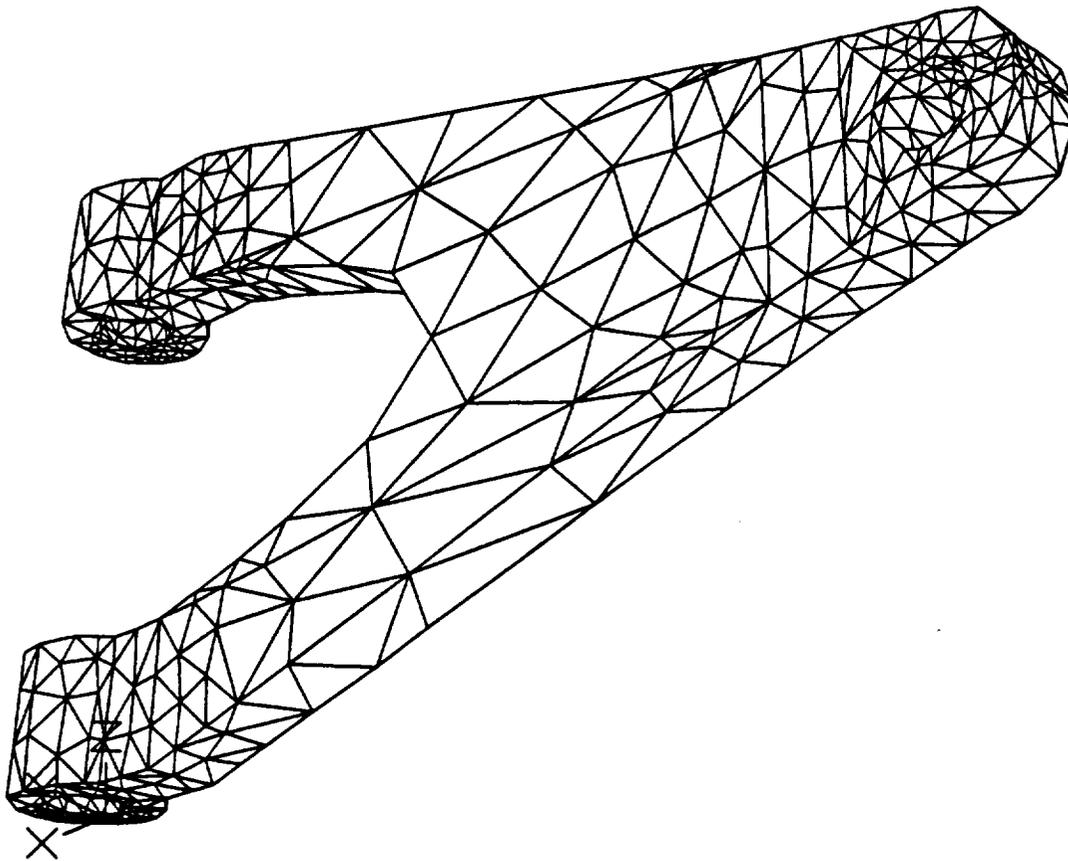**Fig. 1 Mesh-Geometry-Dimension Relationship**
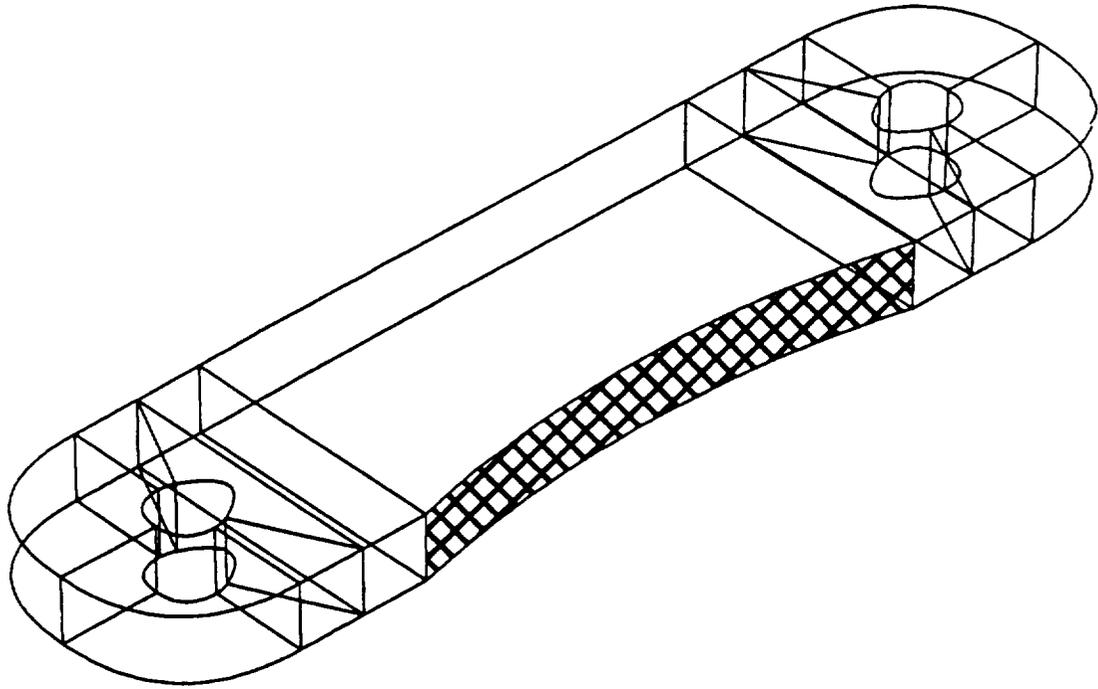
Fig. 2 Torque Arm Mesh

Fig. 3 Control Arm Mesh

Fig. 4 Geometric Model for Torque Arm
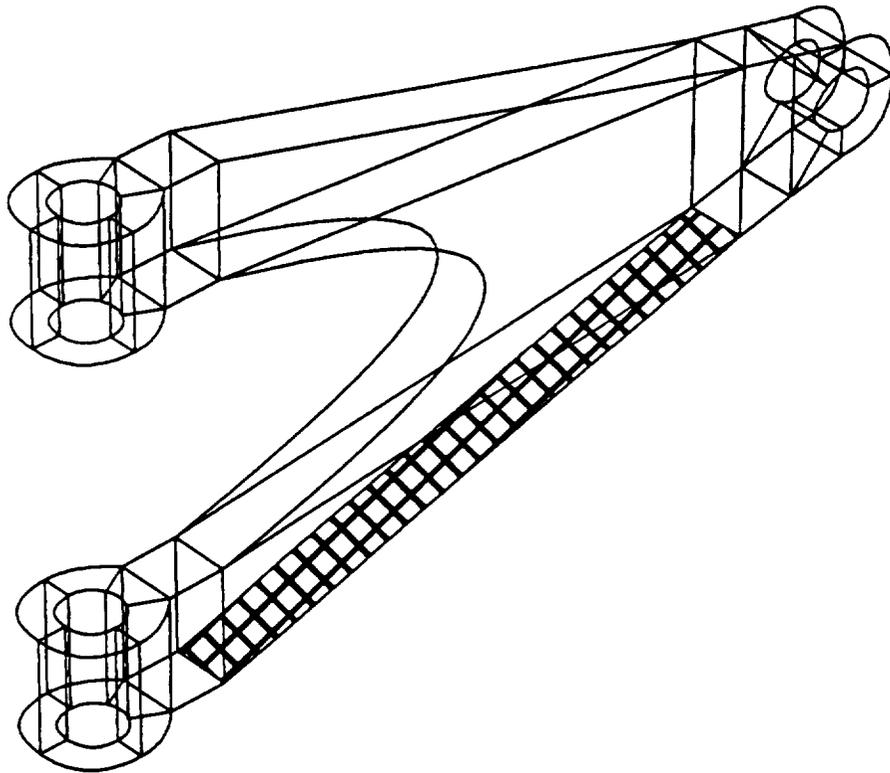

Fig. 5 Geometric Model for Control Arm

215

SHAPE OPTIMIZATION AND CAD

by

JOHN RASMUSSEN

Institute of Mechanical Engineering, Aalborg University
Pontoppidanstraede 101, DK-9220 Aalborg East, DENMARK

Summary: This paper presents a technique for the integration of structural optimization facilities into a Computer Aided Design environment. Problems in connection with the flexibility of the mathematical formulation and the system design are discussed. The techniques have been implemented in the structural optimization system CAOS, and a practical example of the use of this system is presented.

1. Introduction

Structural optimization has attracted the attention since the days of Galileo. Olhoff & Taylor [1] have produced an excellent overview of the classical research within this field. However, the interest in structural optimization has increased greatly during the last decade due to the advent of reliable general numerical analysis methods and the computer power necessary to use them efficiently. This has created the possibility of developing general numerical systems for shape optimization. Several authors, eg., Esping [2], Braibant & Fleury [3], Bennet & Botkin [4], Botkin, Yang & Bennet [5] and Stanton [6] have published practical and successful applications of general optimization systems. Ding [7] and Hörnlein [8] have produced extensive overviews of available systems. Furthermore, a number of commercial optimization systems based on well-established finite element codes have been introduced. Systems like ANSYS, IDEAS, OASIS and NISAOPT are widely known examples.

In parallel to this development, the technology of Computer Aided Design (CAD) has gained a large influence on the design process of mechanical engineering. The CAD technology has already lived through a rapid development driven by the drastically growing capabilities of digital computers. However, the systems of today are still considered as being only the first generation of a long row of Computer Integrated Manufacturing (CIM) systems. These systems to come will offer an integrated environment for design, analysis and fabrication of products of almost any character. Thus, the CAD system could be regarded as simply a database for geometrical information equipped with a number of tools with the purpose of helping the user in the design process. Among these tools are facilities for structural analysis and optimization as well as present standard CAD features like drawing, modelling and visualization tools.

The state of the art of structural optimization is that a large amount of mathematical and mechanical techniques are available for the solution of single problems. By implementing collections of the available techniques into general software systems, operational environments for structural optimization have been created. The forthcoming years must bring solutions to the problem of integrating such systems into more general design environments. The result of this work should be CAD systems for rational design in which structural optimization is one important design tool among many others.

2. The structural optimization system CAOS

A CAD based structural optimization system by the name of CAOS (Computer Aided Optimization of Shapes) has been developed at the Institute of Mechanical Engineering, Aalborg University, Denmark. The purpose of this work is to conduct experiments with various solutions to the CAD integration problems outlined in the preceding section. The widely used commercial CAD system AutoCAD is used as the basis for CAOS, but the system concept is independent of the AutoCAD data structure and the tecniques used in CAOS can therefore be applied in connection with most other CAD systems as well.

CAOS has been under constant development over a period of four years and is today a fully operational shape and topology design system with a number of interesting features. In particular, CAOS offers solutions to the following important problems of CAD integrated shape optimization:

1. There is a large number of possible formulations of the shape optimization problem. One may choose to minimize weight, stress, compliance, displacement or any other property that can be derived from the geometrical model or the output from an analysis program which is usually a finite element module. The same set of possibilities should be available for specification of constraints. Mathematically, these different formulations lead to very different optimization problems.

2. In order to use a mathematical programming technique to solve the problem, the continuous shape of the geometry must be described by a finite, preferably small, number of design variables. This problem is closely connected with the data

structure of the CAD system which is usually not flexible enough to allow for the shape changes required by the optimization module. For instance a circle is defined uniquely by its radius and the coordinates of its center point. In connection with a general optimization system, this data storage scheme has a serious drawback: the circle is bound to remain a circle. It cannot with this data structure be turned into something else, eg., an ellipse. The interface to the optimization system must provide a conversion of CAD data into a form more convenient for shape optimization.

3. The geometrical information is interchanged rather than just passed on, ie., the initial geometry is passed from the CAD model to the optimization application and the optimized geometry goes the opposite way.

4. In most cases, the initial geometry possesses certain measures and shapes that are crucial to the functionality or fabrication and therefore cannot be changed during the shape optimization process. A method must be found to maintain the functionality of the geometry throughout the optimization process.

5. The initial CAD model of the structure has to be converted into a finite element model to be used for the analysis, and this finite element mesh must conform to the changes of the geometry as the optimization process progresses. It is not acceptable for each iteration to call upon the designer to perform a new mesh generation "by hand". On the other hand, automatic redefinitions of the mesh must constantly take the intentions that were laid into the initial mesh generation into consideration. For instance, local concentrations of node points in areas of expected high stress gradients must me preserved.

3. Design model and analysis model

CAOS is based on the important distinction between design model and analysis model. The design model is a variable description of the shape of the structure. It is closely connected with the CAD model and totally distinct from the finite element model that is used for the analysis. The design model consists of so-called design elements as presented by Braibant & Fleury [3]. The design elements have a number of attractive features:

1. They lend themselves to a very easy mesh generation. A number of randomly placed nodes on the boundaries is the only input needed for a complete mesh generation in the design element. Thus, automatic generation of an analysis model based on the current shape of the design model is achieved.

2. The boundaries of the design elements can be curves of almost any character. It is therefore very simple to generate relatively complicated geometries with a small number of design elements. Furthermore, geometrical requirements to the final shape are easily specified by assigning specific curve types to the boundaries in question.

3. The shapes of the boundaries are controlled by a number of master nodes which the boundaries pass through. This creates an evident connection between the design variables (namely the positions of the master nodes) and the shape of the geometry.

4. With the drawing aids of the CAD system it is very easy to draw the design elements in a separate drawing layer on top of the original drawing.

5. Design elements provide a way of liberating the geometrical description from the data structure of the CAD system. Through changes of the boundary shapes, design elements have the capability to represent several different shapes with the same element configuration.

4. Optimization type

The design models of CAOS can be subjected to either topology or shape optimization. The topology optimization is performed by the HOMOPT system developed by Martin Bendsøe which represents an important break-through in structural optimization. HOMOPT works directly on the finite element analysis model generated by CAOS. Furthermore, CAOS has been equipped with facilities for visualization of the optimized topologies that come out of HOMOPT. We shall see later, how the user based on this information can continue with an actual shape optimization. The topology optimization minimizes the compliance of the structure with a bound on the available volume. For further information on the homogenization method, please refer to Bendsøe [9].

The shape optimization is initiated by inserting the necessary specifications, ie. design element definitions, curve types, master nodes etc., into the CAD model. This is all done interactively using the drawing and visualization facilities already

available in the CAD system. This also means that the specifications can be erased, moved, redefined or otherwise modified as any other entity in the CAD model. Based on these specifications, a number of simple text files specifying the optimization problem are generated. The actual optimization is solely based on these files and is therefore independent of the CAD system data structure.

5. Mathematical formulation

The mathematical formulation of the shape optimization problem is as follows:

$$\text{Minimize} \qquad\qquad f(a_i), \qquad i=1..n-1 \qquad\qquad\qquad (1)$$

$$\text{Subject to} \qquad\qquad g_j(a_i) \leq G_j, \qquad i=1..n-1,\ j=1..m \qquad\qquad (2)$$

$$\underline{a}_i \leq a_i \leq \bar{a}_i, \qquad i=1..n-1 \qquad\qquad\qquad (3)$$

where n−1 is the number of so−called design variables, a_i, and m is the number of constraints. (3) are side constraints, i.e., upper and lower limits for the design variables. The functions f and g_j are specified by the user as a part of the optimization specification. They can be picked and combined freely from a library with the following contents:

1. Weight
2. Elastic displacement of a given material point
3. Maximum elastic displacement of any point in the structure
4. Stress (several types) at a given material point
5. Maximum stress (several types) at any point in the structure
6. Compliance

Mathematically, different entries in this list lead to very different optimization problems. Entries 2 and 4 are ordinary scalar functions that can be derived directly from the output of the finite element analysis. Entries 1 and 6 are of integral type and require some postprocessing of the results to be evaluated. Entries 3 and 6 lead to min/max−problems with non−differentiable objective functions.

CAOS makes use of the so−called bound formulation presented by Olhoff [10]. This formulation enables the CAOS system to handle the optimization problem in a uniform way regardless of the blend of scalar−, integral− and min/max−criteria defined by the user.

Given the min/max objective function $f = \max(f_j)$, $j=1..p_o$, and a number of constraints, $g_k = \max(g_{kj}) \leq G_k$, $k=1..m$, $j=1..p_k$, we get the following bound formulation of the problem:

$$\underset{a_i,\ \beta}{\text{Minimize}} \qquad\qquad\qquad \beta \qquad\qquad\qquad\qquad (4)$$

Subject to

$$f_j(a_i) - \beta \leq 0, \qquad j=1..p_o,\ i=1..n-1 \qquad\qquad (5)$$

$$w_k g_{kj}(a_i) - \beta \leq 0, \qquad k=1..m,\ j=1..p_k,\ i=1..n-1 \qquad (6)$$

$$\underline{a}_i \leq a_i \leq \bar{a}_i, \qquad i=1..n-1 \qquad\qquad\qquad (7)$$

An extra design variable, β, has been introduced, rendering the total number of variables to n. By m we designate the original number of constraints regardless of whether these are scalar−, integral− or min/max−functions. The number of points whereby a min/max−condition k is represented, eg. the number of nodal stresses among which the maximum stress is to be found, is termed p_k. This number is obviously 1 if condition k is scalar or integral. The weighting factors w_k are imposed on the constraints to allow them to be limited by the same β−value as f. Prior to the call of the optimizer, w_k is found from the relation:

$$G_t w_t = \beta \rightarrow w_t = \beta / G_t \qquad\qquad (8)$$

The tableau (4) through (7) is valid regardless of the blend of functions f and g_t, and the mathematical operations performed are therefore identical for any problem that the user could possibly define. The problem (4)–(7) is solved by sequential programming using either a SIMPLEX algorithm or the Method of Moving Asymptotes by Svanberg [11].

6. Practical example

The example described here is the optimization of a support beam from a civil aircraft. Further examples of applications of CAOS are available in ref. [12]. The structure in question (fig. 1) has the function of carrying the floor in the fuselage of an Airbus passenger carrier and must meet the following requirements:
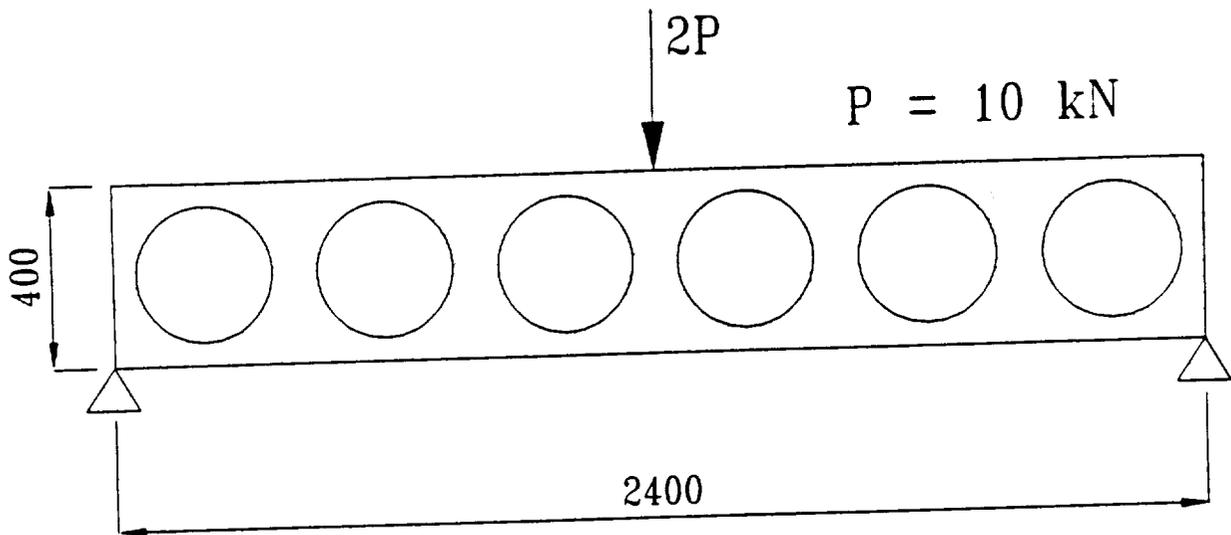

Fig. 1. Initial geometry with loads and boundary conditions.

1. The upper and lower surfaces must be planar and the distance between them cannot be changed.

2. The maximum deflection of the beam must not exceed 9.4 mm under the given load.

3. The maximum von Mises stress must not exceed 385 N/mm².

4. There must be a number of holes in the structure to allow for wires, pipes etc. to pass through. The number, positions and shapes of the holes are free as long as they are of reasonable size.

The purpose of the optimization is to find the design that minimizes the weight of the beam while not violating any of the requirements mentioned above. Because of the symmetry of the structure, we shall analyze and optimize only the right half of the beam. The geometry of fig. 1 has a deflection of 10.1 mm and is thus infeasible by 7.4 %.

We shall initially perform a topology optimization of the structure. As previously explained, this requires a volume bound to be defined. We pick a volume bound of 56% of the full beam and require furthermore the rim of the structure to remain solid. The topology optimization minimized the compliance by distributing the available material optimally within contours of the beam. The resulting topology is shown in fig. 2. It is evident that a number of holes allowing for the necessary passage of wires, pipes etc. have emerged.

We shall now attempt a shape optimization based on this topology. We therefore return to the original definition of the problem, i.e. minimize volume with a bound on displacement and stress. In the creation of the shape optimization model, we shall take the cost of manufacture into account. The complexity of the geometry should therefore be kept at a minimum, i.e., there is a limit to the number of holes that are practical for a structure like this. Furthermore, the upper right corner of the

This part is
considered to
be one hole

This part has
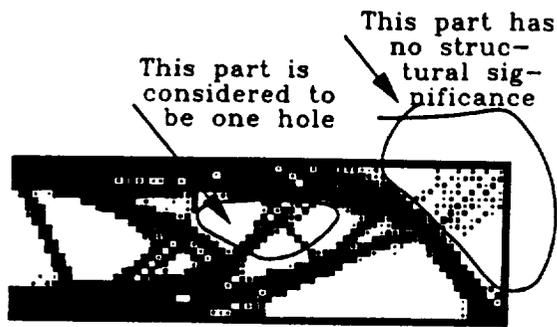no struc-
tural sig-
nificance
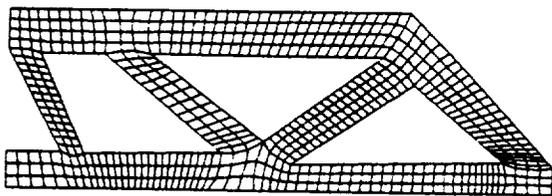
Fig. 2. Result of topology optimization.

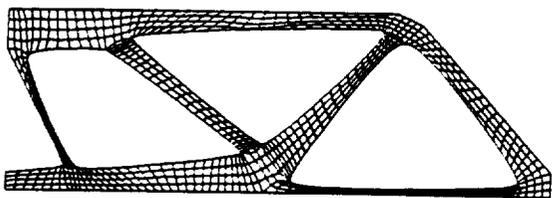Fig. 3. Initial finite element mesh of optimized topology.

Fig. 4. Final finite element model.

frame has been removed. This part of the geometry has a function, but it is structurally insignificant and can therefore be excluded from the shape optimization and added to the modified structure afterwards. This simplification greatly facilitates the generation of the design model. Fig. 2 illustrates the modifications that have been imposed on the optimized topology and the resulting initial finite element model is shown in fig. 3.

CAOS reduces the volume significantly by changing the shape of the the design of fig. 3. The final design is illustrated in fig. 4, based on which the designer can update the CAD model and perform the final adjustments, eg. add the structurally insignificant upper right corner that was removed in order to facilitate the generation of an analysis model, and thereby yield the final design of fig. 5. This design is feasible and the volume is reduced by 42% in comparison with the initial design of fig. 1. Because of the CAD integration, this rather complicated geometry is available directly in the CAD system where the continued design process can take place.

The initial topology optimization allows in many cases the shape optimization to arrive at a much better final result than could otherwise be achieved. A shape optimization of the topology of fig. 1 has been attempted and resulted in a volume reduction of only 5%, ie., for problems like the this, where there are large possibilities for geometrical variations, the topology optimization is a valuable tool in the design process. It is the experience from the present example that topology optimization should be used in the early stages of the development in order to inspire the designer and lead him/her in a beneficial direction. The result of the topology optimization is merely a crude guess and can therefore safely be modified by the designer to meet practical requirements, before the more detailed shape optimization is performed.
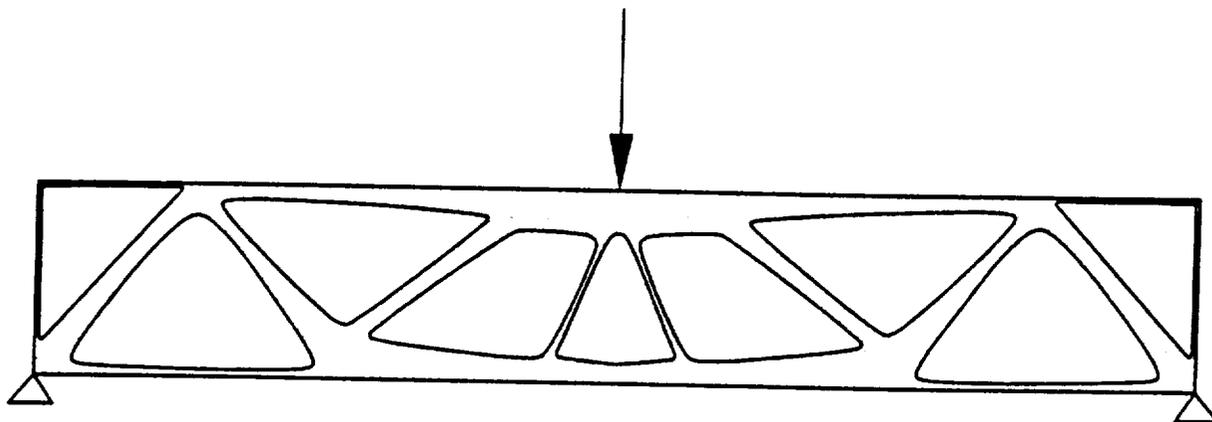

Fig. 5. Example of final geometry slightly modified by the designer. The upper right corner has been added again.

7. Conclusion

The experience from working with CAOS is that the CAD integration of structural optimization is a most valuable tool in the design process. The interactive facilities of the CAD system greatly facilitates the specification of the problem and the optimized geometry is placed directly in the environment where the continued design process will take place.

We have also learned the importance of setting a toolbox of various design facilities at the disposal of the designer. Structural optimization enforces rather than removes the creative aspect of designing, and the final result is therefore very difficult to predict. The collection of structural optimization facilities must be versatile enough to allow the designer to continue work no matter what type of structure emerges. The final design must be a product of creativity rather than availability or lack of analysis facilities.

References

[1]   Niels Olhoff & John E. Taylor: **On Structural Optimization**. Journal of Applied Mechanics, vol. 50, pp. 1139 – 1151 (1983).

[2]   Björn J.D. Esping: **The OASIS Structural Optimization System**. Computers & Structures, Vol. 23, pp. 365–377 (1984).

[3]   V. Braibant & C. Fleury: **Shape Optimal Design using B-splines**. Computer Methods in Applied Mechanics and Engineering, vol. 44, pp. 247 – 267 (1984).

[4]   J.A. Bennett & M.E. Botkin: **Structural Shape Optimization with Geometric Description and Adaptive Mesh Refinement**. AIAA Journal, vol. 23, no. 3, pp. 458 – 464 (1985).

[5]   M.E. Botkin, R.J. Yang & J.A. Bennet: **Shape Optimization of Three-dimensional Stamped and Solid Automotive Components**. James A. Bennet & Mark E. Botkin (ed): The Optimum Shape. Automated Structural Design, Plenum Press, New York (1986).

[6]   E.L. Stanton: **Geometric Modeling for Structural and Material Shape Optimization**. James A. Bennet & Mark E. Botkin (ed): The Optimum Shape. Automated Structural Design, Plenum Press, New York (1986).

[7]   Yunliang Ding: **Compendium. Shape Optimization of Structures: A Literature Survey**. Computers and Structures, vol. 24, no. 6, pp. 985–1004 (1986).

[8]   H.R.E.M. Hörnlein: **Take-Off in Optimum Structural Design**. C.A. Mota Soares (ed.): Computer Aided Optimal Design. Structural and Mechanical Systems. NATO ASI Series F: Computer and System Sciences, Vol. 27, Springer Verlag (1987).

[9]   Martin P. Bendsøe: **Optimal shape design as a material distribution problem**. Structural Optimization, vol. 1, no. 4, pp. 193–202 (1989).

[10]   Niels Olhoff: **Multicriterion structural optimization via bound formulation and mathematical programming**. Structural Optimization, vol. 1, no. 1, pp. 11–17 (1989).

[11]   K. Svanberg: **The Method of Moving Asymptotes – a new method for structural optimization**. International Journal for Numerical Methods in Engineering, vol. 24, pp. 359–373 (1987).

[12]   John Rasmussen: **Collection of Examples – CAOS optimization System, 2nd. edition**. Special report no. 1c, Institute of Mechanical Engineering, Aalborg University, June 1990. ISBN 87-89206-02-9.

A Geometric Representation Scheme Suitable for Shape Optimization

Daniel A. Tortorelli

General Motors Advanced Engineering Staff
Warren, Michigan 48090-9010

Abstract

A geometric representation scheme is outlined which utilizes the natural design variable concept. A base configuration with distinct topological features is created. This configuration is then deformed to define components with similar topology but different geometry. The values of the deforming loads are the geometric entities used in the shape representation. The representation can be used for all geometric design studies; it is demonstrated here for structural optimization. This technique can be used in parametric design studies, where the system response is defined as functions of geometric entities. It can also be used in shape optimization, where the geometric entities of an original design are modified to maximize performance and satisfy constraints. Two example problems are provided. A cantilever beam is elongated to meet new design specifications and then optimized to reduce volume and satisfy stress constraints. A similar optimization problem is presented for an automobile crankshaft section. The finite element method is used to perform the analyses.

1   INTRODUCTION

A shape representation scheme is outlined which utilizes the natural design variable concept. It is ideally suited for modeling objects with similar topology but different geometry. This approach is explored in [1, 2, 3, 4, 5] where the values of loads, used to deform the body, are the design variables in the shape optimization; hence, the terminology *natural* design variables. The method is suited for all geometric design applications; and is demonstrated here for structural optimization.

Shape representation is a concern in parametric design studies. In these studies, a base design is created which is characterized by a series of fixed topological features, (e.g. holes and fillets) and variable geometric entities (e.g. dimensions). An analysis technique, derived from the geometric entities, is formulated to evaluate the system's response. The response for new designs with varied geometries is then readily determined. For example, consider a cantilever beam with transverse tip load P. In this problem, the dimensions: length l, width w, and height

h, comprise the geometric entities; and the maximum bending stress is defined through the formula $\sigma = \frac{6Pl}{wh^2}$. When the finite element method is used as the analysis tool, an automatic mesh generator program is developed which creates the input deck. This approach is presented in [6] for piston design. Creation of these mesh generators can be time consuming. Furthermore, large geometric variations or changes in topology necessitate modifications to the mesh generation program; these modifications may also be time consuming.

Detailed reviews of structural shape optimization methodologies appear in [7, 8, 9, 10]. In these articles it is noted that geometric representation comprises a critical aspect of the shape optimization problem. A subset of the geometric design entities are chosen as the design variables in the optimization. The remaining entities are fixed to meet design and manufacturing requirements. The design variables are altered to improve performance (usually volume dependent) and satisfy constraints (typically stress and displacement dependent). For example, in the cantilever beam problem one may wish to minimize volume and limit the maximum bending stress to a percentage of the yield strength. In this case, only two of the three geometric entities that describe the beam are used as design variables. The span is fixed by the design requirements, so the length is invariable. Only the height and width are varied to obtain the optimum shape.

Several means for selecting the geometric design variables have been proposed when the finite element method is used to perform the analyses required by the shape optimization algorithm. A convenient choice for these variables are the elements of the node coordinate vector. However, as noted in [13], this vector contains too many elements which leads to convergence problems in optimization algorithms, geometric discontinuities over the boundaries, and poor quality finite element meshes. To alleviate this problem, hierarchical parametrization methods have been suggested. In these methods the node coordinates are related to a small number of control points, for instance see [11, 12, 13, 14]. The control point coordinates are then used as the design variables. Unfortunately, creation of the hierarchical design structures can be tedious for large finite element models. Currently, commercial finite element pre-processors offer no capabilities to assist the engineer with this task.

Recently, natural design variable concepts have been

introduced for shape representation [1, 2, 3, 4, 5]. In these methods, the original design configuration is subject to a set of *design loads* and a *shape change analysis* is performed. The elements of the design load set, i.e. the load parameters, are the design variables used in the optimization; and the deformation produced by these loads transforms the original structure into the optimal shape. To meet geometric constraints for design and manufacturing specifications (eg. fixed surfaces, planar surfaces remaining planar, or cylinders remaining cylinders) additional loads in the form of displacement boundary conditions are applied; and linear multi-point-constraints are used. In this way, when the finite element method is used to perform the analyses, commercially available preprocessors can be used to create the design data. Internal mesh distortions introduced from this natural design variable approach are observed to be less severe than those resulting from purely geometric methods [10].

To maintain boundary smoothness requirements, Rajan and Belegundu [3, 4] lay second-order elements over the existing structure when performing the shape change analysis. Yao and Choi [5] pave the bounding surfaces with Bezier surfaces in a manner similar to that suggested in [14] to retain surface regularity. In this latter technique, movements of all surface nodes are linked to the motion of the control points. Prescribed displacements, used as the design variables, are then enforced at the control points to deform the structure. Mathematical relationships are used to relate the displacements of the control points to the surface points. By using prescribed displacements as the design variables rather than forces, the effects that the design loads have on the structural shape are more readily interpreted by the designer.

In some cases, the design loads can be accumulated throughout the optimization process [5]. In [2, 3, 4] this accumulation is not performed. Instead, after each design iteration during the optimization, the structure is redefined by the deformed configuration resulting from the shape change analysis; and the design loads are zeroed. This can be thought of as an *updated-Lagrangian shape representation* If the shape change problem is linear, this practice of resetting the design variables leads to unnecessary computations, as will be seen later.

In this paper, a detailed development of the natural design variable concept is presented as it applies to parametric design studies and shape optimization. The methodology follows that of in [1, 2, 4, 3, 5]. Here, as in [5], the design variables are not reset after each design iteration, i.e. a *total-Lagrangian shape representation* is presented. In this way, great computational saving are obtained if the shape change problem is linear. Thermal loads are introduced into the design load field to obtain localized shape changes. For example, consider the cantilever beam problem. A global deformation of the entire beam is produced if transverse design loads are applied at the free end. Ideally, we would like to obtain localized shape changes of the tip region to optimize the body.

This localized deformation can be obtained from thermal expansion which results from heating or cooling the tip region. Boundary regularity is maintained by using the methods in [1, 4, 3, 5]. Geometric constraints are enforced with essential boundary conditions and nonlinear multi-point-constraints, thus surfaces described by cylindrical, spherical, and toriodal surfaces are retained. To incorporate nonlinear multi-point constraints the sensitivity derivations that appear in [15] are utilized. The treatment of nonlinear shape change problems extends the works of [1, 2, 4, 3, 5].

In the following section, the shape representation scheme is presented as it applies to parametric design studies and optimal design algorithms. Then, two example problems are provided. In the first, a cantilever beam is elongated and then optimized to reduce volume and satisfy stress constraints. In a similar problem, an existing automobile crankshaft is optimized. The finite element method is used to perform the analyses.

2 SHAPE REPRESENTATION CONCEPT

In this section, the shape representation scheme is presented. Initially, it is described in the context of the optimal design process. Then, it is specialized for parametric design studies. Some general remarks regarding the algorithm are also supplied.

2.1 OPTIMAL DESIGN ALGORITHM

In the optimal design process, two configurations of a body Ω with bounding surface Γ are considered, the *base* configuration χ_{κ^b} and the *design* configuration χ_{κ^d}. $\mathbf{X}^b$ and $\mathbf{X}^d$ denote the places that a particle $X \in \Gamma$ occupies in the base and design configurations, respectively. The base configuration remains fixed and coincides with the body Ω; so $\mathbf{X}^b$ is the position vector of the material particle X. The design configuration is obtained through the deformation that results from an application of the of *design load field* $\mathcal{F}^d = [\mathbf{b}^d, \mathbf{E}^{Id}, \mathbf{S}^{Id}, \mathbf{u}^{pd}, \mathbf{s}^{pd}]$

$$\mathbf{X}^d = \mathbf{X}^b + \mathbf{u}^d \qquad (1)$$

$\mathbf{b}^d$, $\mathbf{E}^{Id}$, and $\mathbf{S}^{Id}$ denote the design body force, initial strain, and initial stress acting in Ω; and $\mathbf{u}^{pd}$ and $\mathbf{s}^{pd}$ represent prescribed displacement and prescribed surface traction acting on Γ. $\mathbf{u}^d$ is the resulting *shape displacement field*.

We consider two analyses in this development. The shape displacement field $\mathbf{u}^d$, is determined by a *shape change analysis*. Once $\mathbf{u}^d$ is known, equation (1) is used to define the design configuration χ_{κ^d}. Next, the *real analysis* is performed. This is the conventional analysis performed by the engineer. The *real load field* $\mathcal{F}^r$ is applied to the current design χ_{κ^d} and the *real response field* $\mathcal{S}^r$, of the system is evaluated. Thus, χ_{κ^d} is the

223

undeformed (original) configuration for the real analysis. Once S^r is known, the cost and constraint functionals for the design are determined on χ_{κ^d}. The values of these functions are the components of the *constraint vector field* G.

Conservation of mass requires the density of the real configuration to vary as $\mathcal{F}^d$ is modified. Here, we consider this variation when performing the shape change analysis. However, before we perform the real analysis we equate the density of χ_{κ^b} to that of χ_{κ^d}; i.e., we use a constant density in χ_{κ^d} for all real analyses. If this were not the case, the real body forces and mass would remain constant for all design configurations. Clearly, this situation must not be allowed.

In a finite element implementation of the natural design variable concept second-order elements (e.g. beams and plates), Bezier surfaces, and multi-point constraints may be used to retain design requirements. Second-order elements can be placed on the surface of the body when performing the shape change analysis [3, 4]. In this way, boundary regularity of the deformed finite element mesh (and hence the design configuration) is maintained. These elements must be removed before the real analysis is initiated. Design loads in the form of prescribed displacement boundary conditions, defined through the control point movements of Bezier surfaces, can also be used to maintain boundary regularity [1, 5]. Multi-point constraints may be applied to enforce dimensional requirements. For example, rigid bars can be arranged as the spokes on a wheel of a cylinder to fix the radius, thus allowing translation and rotation of the cylinder. Like the second order elements, these added constraints are to be removed before performing the real analysis. The suggested use of linear multi-point constraints appears in [2]. Here, we use the developments in [15] to extend this idea so that nonlinear constraints may be incorporated.

The real response is an implicit function of the design forces. Variations in $\mathcal{F}^d$, produce alternate design configurations χ_{κ^d}, and ultimately affect the value of G. In a shape optimization, the *design field* is defined to minimize the cost function and satisfy the constraints. Here, the design field is equivalent to $\mathcal{F}^d$.

In this paper, we rely on numerical optimization algorithms to select the optimal load set $\mathcal{F}^d$. These algorithms are restricted to finite-dimensional design spaces. With this motivation, we parameterize $\mathcal{F}^d$ over χ_{κ^b} by an M-dimensional vector ϕ and an N-dimensional vector φ and a set of basis functions. This type of parameterization is used in the finite element method, where the load fields are defined at node points and interpreted locally in each element by shape functions [16]. The parameters in ϕ are used to maintain geometric design and manufacturing constraints. Typically, they define prescribed displacements over selected surfaces; they are held constant during the optimization. The parameters in φ describe loads which locally deform portions of the body into the optimum shape. They are selected by the optimization

algorithm; thus they serve as the design variables. G is written as an L-dimensional vector; its elements consist of the values of the cost function and the $L-1$ constraint functions. The discretized constraints components G_α $\alpha = 2, N$ can be defined by evaluating the constraint field G at a distinct point or averaging the value of G over a sub-region of χ_{κ^d}.

For example, suppose it is desired to optimize a cantilever beam of length l' for minimum mass subject to a constraint on the maximum bending stress. A beam of length l, width w, and height h serves as the base configuration χ_{κ^b}. To ensure the length requirement is met, a prescribed displacement with value $l' - l$ is enforced at the tip face in the longitudinal direction. This value $l' - l$, is given by a parameter in ϕ. Thermal loads are applied to locally deform the beam's height and width. These load values are related to parameters in φ. The design load field $\mathcal{F}^d$ (composed of the prescribed displacement and thermal loads) is applied to χ_{κ^b}; and a shape change analysis is performed to determine the current design configuration χ_{κ^d}, of the beam. Then, the real loads $\mathcal{F}^r$ are applied to χ_{κ^d}; the real analysis is performed to evaluate the real response S^r; and the maximum bending stress is determined. At this point the mass is evaluated over χ_{κ^d}. Next, G is assembled; its components consist of the mass and the maximum bending stress. As the values in φ change, the design configuration χ_{κ^d}, is modified, and the value of G is altered. To determine the φ that will minimize mass and satisfy the stress constraint an optimization algorithm is used.

Most numerical optimization strategies are iterative. The existing design configuration χ_{κ^d}, is modified; and G is re-evaluated on the modified structure. The design sensitivities $\frac{\delta G}{\delta \varphi}$ must also be evaluated for each iteration. Sensitivity information is used by the optimizer to select the direction of the design variation $\delta\varphi$, that will produce the most improved design. The magnitude of $\delta\varphi$ is determined by one of several methods [17]. Once $\delta\varphi$ is known, the design is updated to $\varphi = \varphi + \delta\varphi$; and the next design iteration commences. After several iterations, the magnitude of $\delta\varphi$ becomes small; at this point, the program has converged to the optimized design.

In the present approach, one shape change analysis is performed for each design iteration to determine the design configuration χ_{κ^d}. These time consuming analyses represent one serious drawback of natural design variable method. However, if the shape change problem is linear, only one shape change analysis is required throughout the design process. We use superposition to define current configuration χ_{κ^d} for each subsequent design iteration, i.e.

$$\mathbf{X}^d = \mathbf{X}^b + \mathbf{u}_\phi^d + \sum_{\alpha=1}^{N} \mathbf{u}_{\varphi_\alpha}^d \varphi_\alpha \qquad (2)$$

where $\mathbf{u}_\phi^d$ is the displacement field on χ_{κ^b} obtained by equating φ in $\mathcal{F}^d$ to zero; and $\mathbf{u}_{\varphi_\alpha}^d$ is the displacement

field on χ_{κ_b} obtained by equating φ_α in $\mathcal{F}^d$ to one, and equating all the remaining parameters in ϕ and φ to zero. In all, this requires $N + 1$ analyses to determine the displacement fields defined in equation (2). If the finite element method is used to solve the shape change problem, then this amounts to one stiffness matrix assembly and decomposition followed by $N + 1$ load vector assemblies and back substitutions for *all* design updates. In the updated-Lagrangian shape representation approach, the base configuration is updated after each design iteration. This requires a stiffness matrix assembly and decomposition followed by $N + 1$ load vector assemblies and back substitutions for *each* design update. Thus, in the total-Lagrangian shape representation formulation, significant computational advantages are obtained. Comparisons between finite element meshes created by the updated-Lagrangian and total-Lagrangian methods exhibit subtle differences [5]. Thus, the total-Lagrangian method may be preferred in light of its efficiency. In [5], the base design is updated periodically during the optimization.

Consider the case where we wish to design several components which are characterized the same base configuration but different values of ϕ, i.e. they have different dimensions. If the shape change problem is linear, we express u_ϕ^d as

$$u_\phi^d = \sum_{\beta=1}^{M} u_{\phi_\beta}^d \phi_\beta \tag{3}$$

where $u_{\phi_\beta}^d$ is the displacement field on χ_{κ_b} obtained by equating ϕ_β in $\mathcal{F}^d$ to one, and equating all the remaining parameters in ϕ and φ to zero. This equation requires the solution of M additional analysis. In finite element method implementations, M additional load vector assemblies and back substitutions are required. In all, one stiffness matrix assembly and decomposition, followed by $M + N$ load vector assemblies and back substitutions are performed for the shape change analysis. After this initial analysis is completed, we can readily obtain the design configuration that corresponds to any ϕ - φ combination from equations (2) and (3); no re-analysis is necessary.

To evaluate the sensitivities $\frac{\partial G}{\partial \varphi}$ we follow the approach used in [18, 15]. Therein, explicit sensitivities of a general response functional are derived with respect to the variations of both shape and non-shape design parameters. Domain parameterization is used to derive shape sensitivities [19]. In this technique, an invertable mapping is introduced which relates the undeformed configuration to a fixed reference configuration and a finite set of design parameters. The shape sensitivities are expressed in terms of the explicit variations of the mapping. In [11, 12, 13, 14] the mappings are defined by Bezier and parametric curves and surfaces [20]. Variations of interior line and surface points with respect to control point variations are readily determined for these mappings. Once these explicit variations are known, the shape sensitivity analyses can be performed by following the procedures outlined in [18, 15]. As noted in [15], if the number of design parameters exceed the number of constraint functionals the adjoint sensitivity method requires less analyses than the direct differentiation method and should be used; otherwise the direct differentiation technique is preferred.

When the natural design variable method is used, addition consideration is required to evaluate the shape sensitivities. Here, equation (1) defines the invertable mapping $X^d = X^d(X^b, \varphi)$, that relates the undeformed configuration χ_{κ_b} to the reference configuration χ_{κ_b} and the design parameters φ. The problem arises when we attempt to determine the explicit variations of this mapping. X^d is *implicitly* defined by ϕ and a boundary-value problem; thus the explicit variations of the mapping $\frac{\partial X^d}{\partial \varphi} \delta \varphi$ are not readily available. To determine $\frac{\partial X^d}{\partial \varphi}$ consider the identity

$$\begin{aligned} \frac{\partial X^d}{\partial \varphi} &= \frac{\partial(X^b + u^d)}{\partial \varphi} \\ &= \frac{\partial u^d}{\partial \varphi} \end{aligned} \tag{4}$$

derived from equation (1); recall that X^b is constant. $\frac{\partial u^d}{\partial \varphi}$ is the partial derivative of the shape displacement field with respect to the design vector φ. This term can be obtained by using the direct differentiation method described in [15]. Thus, two sensitivity analyses are required at each design iteration when the natural design variable method is implemented. First, a direct differentiation design sensitivity analysis of the *shape change* problem (defined on χ_{κ_b} with loads $\mathcal{F}^d$) is used to determine $\frac{\partial u^d}{\partial \varphi}$. Once $\frac{\partial u^d}{\partial \varphi}$ is known, a second, adjoint (if $L < N$) or direct differentiation (if $L > N$) sensitivity analysis is performed for the *real* problem (defined on χ_{κ_d} with loads $\mathcal{F}^r$) to determine $\frac{\partial G^d}{\partial \varphi}$.

The fact that two sensitivity analyses are required for each design iteration represents the second serious draw back of the natural design variable method. Fortunately, these sensitivities can be obtained in an efficient manner if the finite element method is used. In the real sensitivity analysis, a minimum of L or N load vector assemblies and back substitutions are required to determine $\frac{\partial G}{\partial \varphi}$. These analyses are performed using the decomposed tangent stiffness matrix from the real analysis. The form of the load vectors and the means for evaluating the sensitivity $\frac{\partial G}{\partial \varphi}$ is given in [15]. However, before these sensitivities are evaluated, $\frac{\partial u^d}{\partial \varphi}$ must be determined. This requires N additional load vector assemblies and back substitutions. These N analyses use the decomposed tangent stiffness matrix from the shape change problem. Each of the N load vectors is obtained by equating φ_α in $\mathcal{F}^d$ to one, and setting all the remaining parameters in ϕ and φ to zero (see [15] for details). The resulting N displacement fields equal the sensitivities $\frac{\partial u^d}{\partial \varphi_\alpha}$, $\alpha = 1, N$.

The natural design variable method requires one shape change analysis to determine the displacement u^d and

one sensitivity analysis to evaluate the sensitivities $\frac{\partial \mathbf{u}^d}{\partial \varphi}$ at each design iteration. In the finite element context, this requires one additional stiffness matrix formulation and decomposition, followed by $N+1$ additional load vector assemblies and back substitutions *in addition to* the real analyses used to evaluate $\mathcal{S}^r$ and $\frac{\partial G}{\partial \varphi}$ at each design iteration. Thus, a significant computational penalty is paid.

This computational penalty is significantly reduced if the shape change problem is linear. As previously noted, one analysis is performed and then superposition is utilized to evaluate $\mathbf{u}^d$ (cf. equation (2)). In addition, we will show that no additional analysis is required to evaluate the sensitivity $\frac{\partial \mathbf{u}^d}{\partial \varphi}$. For this linear problem, the tangent stiffness matrix is constant, i.e. independent of $\mathbf{u}^d$; and from [15] it is seen that

$$\frac{\partial \mathbf{u}^d}{\partial \varphi_\alpha} \equiv \mathbf{u}^d_{\varphi_\alpha} \qquad \alpha = 1, N \qquad (5)$$

where $\mathbf{u}^d_{\varphi_\alpha}$ are the displacement fields which have already been evaluated in equation (2). [1] Thus, no shape change sensitivity analysis is required. In all, to evaluate the deformed configuration χ_{κ_d} and sensitivities $\frac{\partial \mathbf{u}^d}{\partial \varphi}$ for *all* design iterations, only *one* stiffness matrix assembly and decomposition, followed by a *total* of $M + N$ load vector assemblies and back substitutions are performed. The current design configuration is obtained from equations (2) and (3); and the sensitivities are evaluated from equation (5). Further, *no* additional shape change analyses are required to optimize designs with different ϕ values; i.e. only *one* shape change analysis is performed to optimize any design created from χ_{κ_b}. For the linear shape change problem only a small computational penalty is paid when we incorporate the natural design variable technique with the total-Lagrangian shape representation. In the updated-Lagrangian shape representation, this computational savings is not realized.

We are now in position to outline the optimization algorithm which utilizes the sequential linear programming strategy [17]. Initially, a base design configuration χ_{κ_b} is chosen; the geometric constraints ϕ are specified; a starting value of the design variables φ is selected; and the real load field $\mathcal{F}^r$ is supplied. For each design iteration, the design load field $\mathcal{F}^d$ is assembled; a shape change analysis is performed to evaluate the shape displacement field $\mathbf{u}^d$ and the sensitivities $\frac{\partial \mathbf{u}^d}{\partial \varphi}$; the current design configuration χ_{κ_d} is updated from equation (1); the real response $\mathcal{S}^r$ is determined; the constraint vector G and sensitivities $\frac{\partial G}{\partial \varphi}$ are evaluated; and the optimizer is called. The design parameters in φ are modified by the optimizer and the process is repeated until the design converges. If the shape change problem is linear, then the shape change analysis is initially performed, rather than at each design iteration, to evaluate the displacement fields $\mathbf{u}^d_{\phi_\beta}$, $\beta = 1, M$

and $\mathbf{u}^d_{\varphi_\alpha} = \frac{\partial \mathbf{u}^d}{\partial \varphi_\alpha}$, $\alpha = 1, N$. At each design iteration, the current design configuration χ_{κ_d} is updated from equations (2) and (3).

2.2 PARAMETRIC DESIGN STUDY

In a parametric design study, a portion of the aforementioned algorithm is used. The base design configuration χ_{κ_b}, is supplied and is subsequently deformed by applying the load set $\mathcal{F}^d$. ϕ is chosen to ensure that the appropriate geometric design and manufacturing constraints are satisfied. The elements of φ comprise the design parameters. For each φ, the design load field $\mathcal{F}^d$ is assembled; a shape change analysis is performed; the current design configuration χ_{κ_d} is updated from equation (1) [2]; the real response $\mathcal{S}^r$ is determined; and the constraint vector G is evaluated. Plots or tables of performance verses design parameters are then used to choose possible design candidates. No sensitivity information is required.

When the finite element method is used to perform the analyses, this procedure creates the finite element mesh for each of the varied designs. For this reason, it is viewed as a mesh generator [1]. If large geometric changes or topological changes are desired, then the base design must be altered. If a mesh generator is used, additional coding is required to reflect any such changes. Most likely, changes to the base design can be performed more easily than changes to the mesh generation program.

2.3 SIDE NOTES

The format of the optimal design algorithm is not unique. Here, sequential linear programming is utilized. Other possibilities are sequential quadratic programming or the modified method of feasible directions [17]. In these methods, an inner 1-dimension search loop is used to determine the magnitude of the design variation vector $\delta\varphi$. In this loop, the the design variables are modified and the value of the constraint vector G is re-evaluated; no gradient information is requested. Often, rather than performing a complete reanalysis to determine G, efficient approximate problems are created to predict these values, for example see [21].

The means for evaluating the shape sensitivities $\frac{\partial G}{\partial \phi}$ is not unique. The material derivative approach [22] or semi-analytical approach [23] are also viable options. All of these approaches yield identical results when used correctly.

Several techniques have been suggested to reduce computational requirements. In [3] it is noted that softer material properties for the shape change analysis may lead to faster convergence of the optimization. It is suggested in [1, 5] to use substructuring when performing the shape

---

[1] These sensitivities are consistent with those appearing in [2].

[2] If the shape change problem is linear, the displacement fields $\mathbf{u}^d_{\phi_\beta}$, $\beta = 1, M$ and $\mathbf{u}^d_{\varphi_\alpha}$ $\alpha = 1, N$ are evaluated once; and the current design configuration χ_{κ_d} is updated from equations (2) and (3).
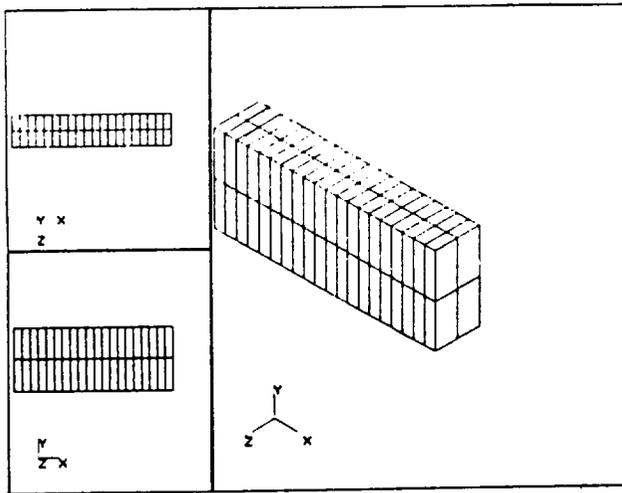
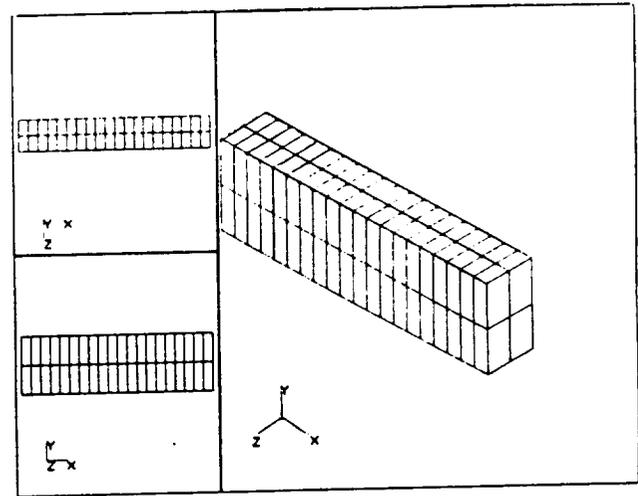Figure 1: Cantilever beam base design.

Figure 2: Cantilever beam initial design.

change analysis. In this way, the shape change analyses can be performed more efficiently if a portion of the finite element model is held fixed during the design process.

Finally, note that this methodology is not restricted to the finite element method, or structural problems. Any means may be used to evaluate the system's response, and the response for any system which is governed by geometric entities (e.g. heat transfer and fluids problems) can be optimized or studied in a parametric manner.

3 EXAMPLES

Two example problems are presented in this section. A cantilever beam is elongated to meet new design specifications; and then it is optimized to reduce volume and satisfy stress constraints. In a similar problem, an automobile crankshaft is optimized. In all cases, the displacement-based finite element method is used to perform the analyses; and sequential linear programming is used to perform the optimization.

3.1 CANTILEVER BEAM

Consider the finite element model of a cantilever beam shown in figure 1. The beam is fixed at the wall and subjected to a distributed transverse tip load of 10,000N. The isotropic beam has a Young's modulus of 1.0×10^7MPa and a Poisson's ratio of 0.3. The length, width, and height of the beam is 50mm, 10mm, and 20mm, respectively. The finite element model consists of 80 8-node bilinear hexahedrons, 189 nodes, and 540 degrees of freedom. Current design requirements limit the maximum allowable bending stress magnitude to 3000 MPa. A similar problem appears in [24, 11].

For the current design problem, the beam is required to be 60mm long and 10mm wide; only the height is al-

lowed to vary. The existing finite element model, i.e. the $50 \times 10 \times 20$mm beam, shown in figure 1, is used as the base design configuration χ_{κ^b}. To meet the geometric design constraints, prescribed displacements with magnitude 10mm are applied in the positive x-direction to the free end of the beam; the side surfaces are constrained in the z-direction; the fixed end is constrained in the x-direction; and the bottom edge of the fixed end is constrained in the y-direction. The parameters describing these prescribed displacements are the elements in ϕ. The first parameter $\phi_1 = 10 = 60 - 50$, defines the beam's length; the second parameter $\phi_2 = 0 = 10 - 10$, defines the width.

The design variables in φ describe thermal loads that are applied during the shape change analysis. Moving from left to right, eleven regions B_μ^b, $\mu = 1, 11$ which consist of elements 1-8, 9-16, 17-24, 25-32, 33-40, 41-48, 49-56, 57-64, 65-72, 73-76, and 77-80 are defined. [3] Each region is characterized by distinct coefficients of thermal expansion; thus each region is subjected to different thermal strains E^{Id}. The eleven coefficients comprise φ. The temperature of the body is uniform, 1°C. Initially, φ is zero; ultimately, it is defined by the optimization algorithm.

The initial design configuration is shown in figure 2. This beam is slightly thinner than the base configuration due to the Possion's effect. Results of the real analysis in which the initial design configuration is subjected to the tip load are shown in figure 3. This contour plot depicts the bending stress magnitude. As seen from this figure, the beam is greatly over designed.

To reduce the beam's volume a shape optimization is performed. The objective function is the volume and the constraints are the normalized bending stress less one (a

---

[3]( )^d and ( )^b denote quantities defined in the design configuration χ_{κ^d} and base configuration χ_{κ^b}, respectively.

Figure 3: Cantilever beam initial design stress contours.
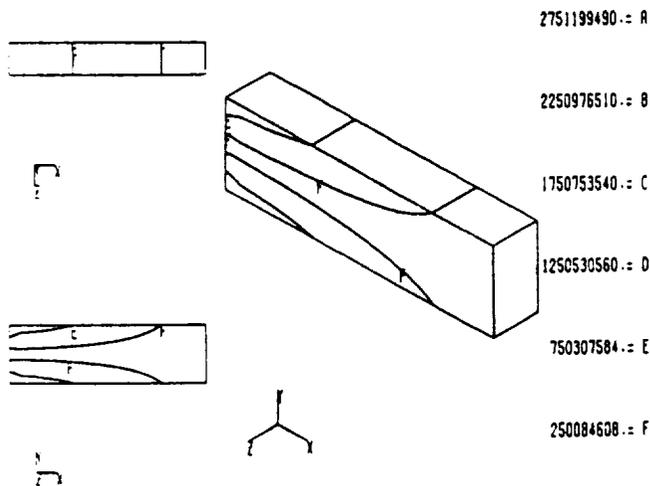

Figure 4: Cantilever beam optimal design.

negative constraint value denotes constraint satisfaction). The stress is evaluated at the nodes using nodal averaging techniques. We use symmetry to define the stress constraint locations. The node positions along the top surface, specifically the center line and one edge, of the beam are chosen; a total of 42 constraints are defined.

A linear shape change analysis is performed for this problem. Before the first design iteration the stiffness matrix used for the shape change analysis is assembled and decomposed once. The displacement fields $u^d_{\phi_\beta}$, $\beta = 1, 2$ and $u^d_{\varphi_\alpha} = \frac{\partial u^d}{\partial \varphi_\alpha}$, $\alpha = 1, 11$, are computed as discussed in the last section. A total of thirteen load vector assemblies and back substitutions are required.

For each design iteration, equations (2) and (3) are used to update the current design configuration. A real analysis is performed over χ_{κ^d} to evaluate the real response resulting from the tip load. The constraint vector G and sensitivities $\frac{\partial G}{\partial \varphi}$ are subsequently evaluated. Finally, the optimizer is called; and φ is updated.

Results of the optimization appear in figures 4 - 6. These figures depict the optimized configuration, the stress contours at the optimal configuration, and the design history. The top and bottom surfaces, with the exception of the base and tip regions, are maximally stressed. The volume is reduced from an initial value of 10.97142 cm^3 to 4.38744 cm^3. 9 design iterations were required, although convergence was reached after iteration 7. A total of 93.28 CPU seconds were required on a single processor of a CRAY XM-P computer. A majority of the computational expense was spent computing the shape sensitivities $\frac{\partial G}{\partial \varphi_\alpha}$ $\alpha = 1, 11$. This result closely resembles the theoretical solution given in [11].

Suppose we now wished to optimize a beam 70mm long and 15mm wide. No additional shape change analyses would be performed. ϕ_1 and ϕ_2 are equated to 20mm
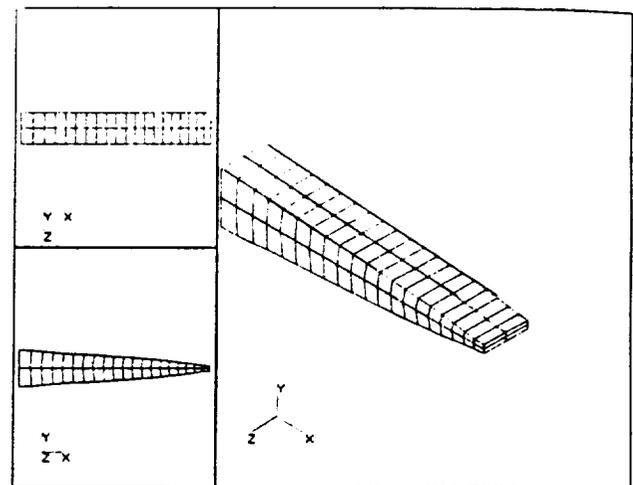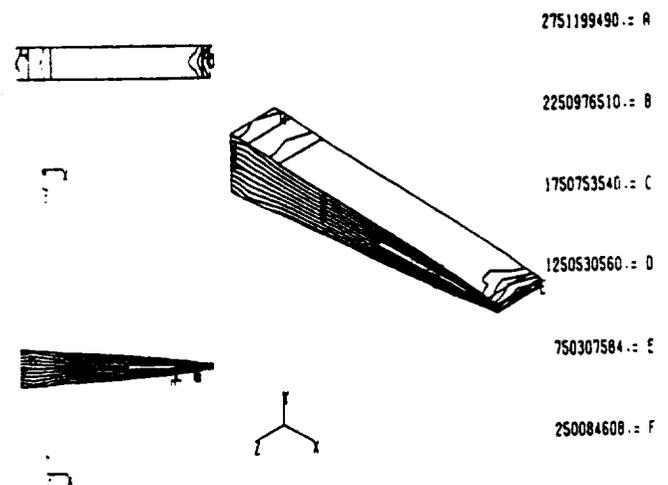

Figure 5: Cantilever beam optimal design stress contours.

OBJECTIVE HISTORY

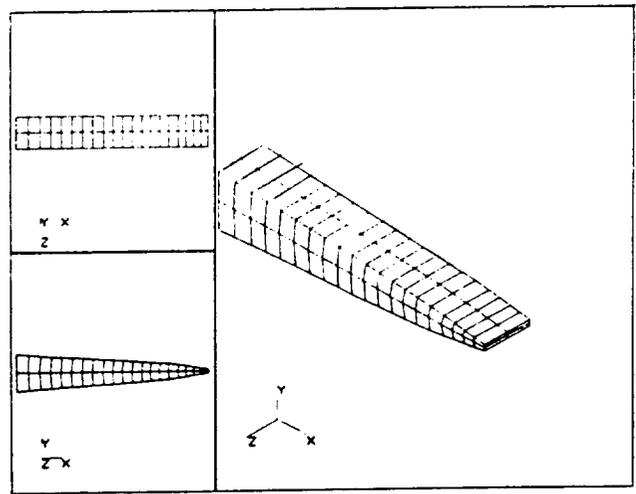Figure 7: Constrained cantilever beam optimal design.

CONSTRAINT HISTORY

Figure 6: Cantilever beam optimization history.

and 5mm, respectively; and the optimization process is repeated. It may be advisable to initialize the starting value of φ for the 70mm beam optimization to the optimized value of φ obtained from the 60mm beam design. This initial guess may lead to quicker convergence, hence reduced computational expense.

As a second exercise, the same problem is optimized. However, this time, additional geometric manufacturing requirements are specified that restrict the beam's top and bottom faces to consist of four piece-wise planar surfaces. One way to maintain this piece-wise planar condition is to apply nonlinear multi-point-constraints to the previous shape change problem. The form of these constraint equations are given in the appendix. Figures 7 - 9 are analagous to figures 4 - 6 for the first problem. In this latter problem, the optimized volume is slightly greater, 4.39131 cm³, possibly due to the shape restriction. The optimization required 8 cycles, although convergence was reached after 6. Computational requirements for this problem were considerably greater, 140.1s even though fewer iterations were required. The increase is attributed to the nonlinear shape change problem. [4] Several Newton-Raphson iterations are required to compute the shape displacement $\mathbf{u}^d$ at each design iteration; then eleven additional load vector assemblies and back substitutions are performed to evaluate the sensitivities $\frac{\partial \mathbf{u}^d}{\partial \varphi_\alpha}$, $\alpha = 1, 11$.

As an alternative, we could have eliminated the nonlinear multi-point constraints by carefully applying prescribed displacements over the planar surfaces. These displacements would be prescribed at the plane interfaces and vary linearly over the adjacent planes (in a manner similar to that in which prescribed displacements are en-

---

[4]Nonlinearities are introduced via the multi-point-constraints. All remaining relationships are linear (eg. stress-strain, strain-displacement).
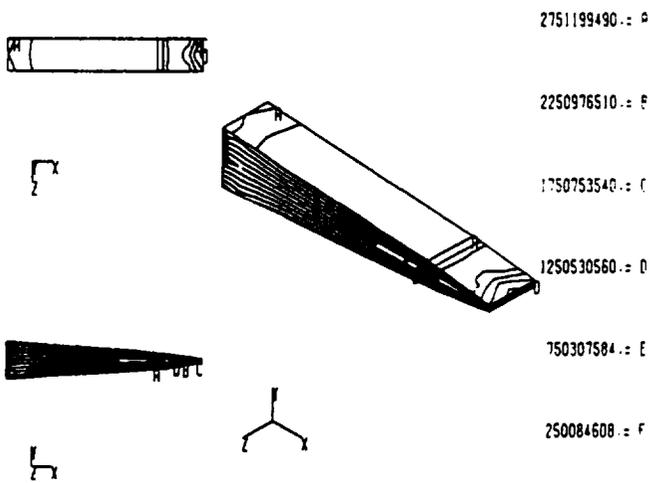
229

2751199490.: A

2250976510.: B

1750753540.: C

1250530560.: D

750307584.: E

250084608.: F

Figure 8: Constrained cantilever beam optimal design stress contours.

58.8: A

45.6: B

32.5: C

19.3: D

6.17: E

-6.98: F
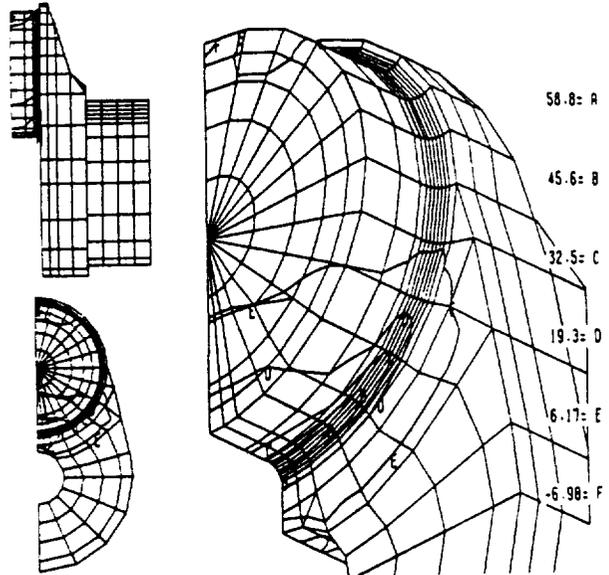
Figure 10: Automobile crankshaft original design.

OBJECTIVE HISTORY

ITERATION NUMBER

CONSTRAINT HISTORY
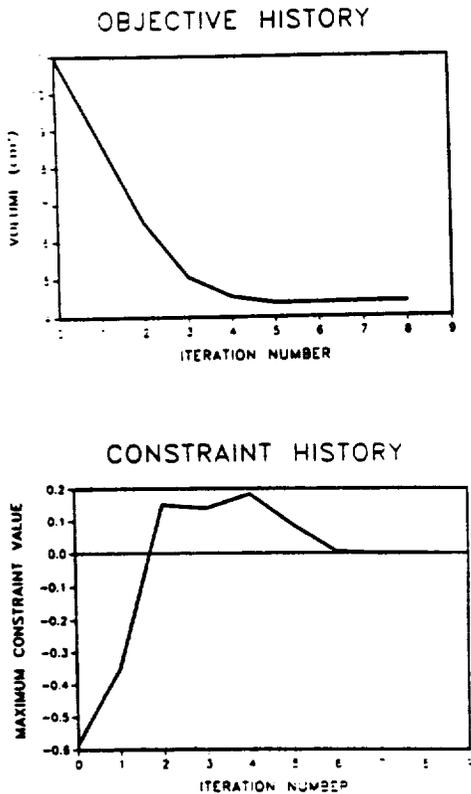
ITERATION NUMBER

Figure 9: Constrained cantilever beam optimization history.

forced over linear finite elements). In this case, the shape change problem is linear and the values of the displacements represent the design variables.

3.2 AUTOMOBILE CRANKSHAFT

In this problem, an existing automobile crankshaft is redesigned to reduce weight and maintain the original design's stress levels. To perform the analyses, the natural design variable concept has been implemented in the general purpose GM optimization code, ODYSSEY. The shape optimization portion of this program is similar to that of the SHOP3D program [26].

The original design and response is illustrated in figure 10. Symmetry has been used to reduce computations. The model consists of 1357 linear solid elements and 1636 nodes. The contour plot illustrates the major principal stress under a bending load. The load results from the gas forces that are transmitted to the pin journal via the piston and connecting rod. The main bearing journal (with the cored section) is constrained by the engine block and bearing caps.

Only the arm section geometry is allowed to vary during the optimization. All three degrees of freedom of the main and pin journals and the refined fillet area and the radial degrees of freedom of the core are fixed by applying homogeneous boundary conditions. The bottom width of the arm is also fixed, as a counterweight is attached to this region. The shoulder region around the pin journal is fixed by constraining the fillet region. The shoulder region around the main journal is similarly constrained.

The design loads consist of 42 thermal load sets and 2 prescribed displacement sets. All of the load sets were

230

quickly created by using an interactive finite element pre-processor program. The thermal loads are defined by applying unit nodal temperatures to sub-regions of the arm. The thermal expansion coefficient is constant throughout the crankshaft. Unit prescribed displacements are applied to the surface nodes opposite the plane of symmetry in the front view (lower left) in the transverse direction. Only those nodes which are not attached to the two shoulders are loaded. All the nodes in the axial direction are given the same prescribed displacement. In all, two displacement sets are defined. The nodal temperature values and the prescribed displacement values are used as the design variables and adjusted during the optimization. In the shape change analysis, one stiffness matrix decomposition and 44 back substitutions are performed. The analysis is linear and hence, it is performed only once.

The constraints consist of the maximum nodal major principal stress for each element, except those adjacent to the load application point. In all 1350 constraints were defined, however only 5-6 were active during the optimization. Eight optimization iterations were performed, after which several elements experienced undesirable distortions so the optimization was terminated.

The results of the optimization are seen in figure 11. The total mass reduction is 8.89%. This is computed over the entire model, so the arm mass reduction is considerably higher. The maximum stress was also slightly reduced. It is seen that mass is added opposite the lower pin and subtracted opposite the upper pin and main. It is also noted that the width of the arm is reduced. However, it is believed that this arm reduction will lead to an increased torsional vibration. Ideally, the optimization should account for both the bending and torsional load cases.

The arm surface opposite the pin is fairly rough. This area could be smoothed by adding more temperature load sets in this region or by using prescribed displacements defined from Bezier surfaces as in [5]. The ability to apply these prescribed displacements is somewhat limited as we have access to only the finite element model data. Nevertheless, the trends obtained from this exercise give designers invaluable information.

4 Summary

A shape representation scheme has been presented and fully detailed. The method uses the natural design variable concept which has been previously introduced in the literature. The total-Lagrangian shape representation is incorporated which offers distinct advantages over the previously used updated-Lagrangian representations if the shape change problem is linear. The representation can be used for shape optimization and parametric design studies; and it is applicable to any problem which is governed by geometric entities.

When the finite element method is used to perform the analyses, then the shape change problem can be easily
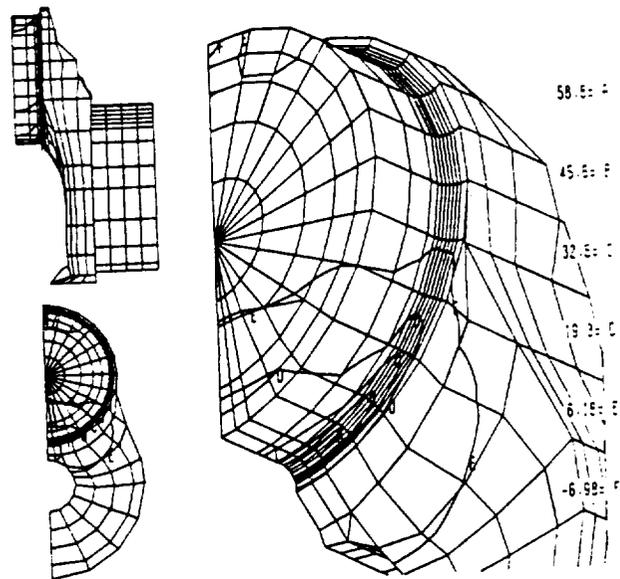

Figure 11: Automobile crankshaft optimal design.

created by applying additional load sets to the existing finite element mesh. These load sets can be created by using commercially available finite element pre-processors. On the other hand, if the shape representation relies on hierarchical parametrization methods, then the design model may be more difficult to create as no commercially available software has been developed for this purpose. For linear shape change problems, the additional cost to implement the shape representation amounts to one stiffens matrix formation and decomposition followed by numerous load vector assemblies and back substitutions.

In the example problems, thermal loads were applied to deform the base configuration into the optimally shaped design. Thermal loads were chosen to create a localized deformation in the structure. In the cantilever beam problem, the optimized designs appeared to be maximally stressed and convergence was reached after a few iterations. It was seen that the nonlinear shape change problem required a considerable amount of additional CPU time as compared to the linear shape change problem. In the automobile crankshaft problem, mesh distortions led to an early termination. Nonetheless, the modified design was considerably improved.

5 ACKNOWLEDGEMENTS

6 APPENDIX

A constraint equation which constrains an independent point to lie on a plane defined by three dependent points is defined as follows.

Initially the points X_α, $\alpha = 0, 3$ are coplanar. X_0 is the dependent point and lies on the plane defined by the three independent non-collinear points. Each point displaces by the amount u_α, $\alpha = 0, 4$ under a deformation and the new point locations are defined $X'_\alpha = X_\alpha + u_\alpha$, $\alpha = 0, 3$.

After the deformation, the point X'_0 is constrained to lie on the plane defined by the points X'_α, $\alpha = 1, 3$. This constraint is met if the the following equation is satisfied [25]

$$h(u) = 0 = n' \cdot V'_{10} \qquad (6)$$

where $V'_{\alpha\beta} = X'_\alpha - X'_\beta$, $\alpha, \beta = 0, 4$ and $n' = V'_{12} \times V'_{13}/(|V'_{12}| |V'_{13}|)$. $\cdot$ denotes the dot product; $\times$ denotes the cross product; and $| |$ denotes the magnitude.

The Lagrange multiplier method is used [16] and $\int_B phdv$ is augmented to the generalized potential energy function, where p is the Lagrange multiplier. The stationary condition of the augmented functional with respect to admissible variations of u and p yields the solution to the boundary-value problem. This problem is nonlinear due to the form of h. Sensitivities for this type of problem appear in [15].

In the cantilever beam example problem, each plane consists of eighteen node points. Three nodes are chosen as the independent points. fifteen constraint equations are defined to ensure the remaining nodes will lie on the plane after the shape change analysis. The element faces defined by these nodes remain planer due to the form of isoparametric parametric mapping. In all one-hundred constraint equations are defined, some equations along common edges are redundant, thus they are eliminated. Additionally, the nodes along the bottom edge of the fixed end are not constrained because of the applied boundary conditions.

References

[1] K.K Choi, T.M. Yao, 3-D Modeling and Automatic Regridding in Shape Design Sensitivity Analysis, Sensitivity Analysis in Engineering, NASA Conference Publication 2457, pp. 329-345, 1987.

[2] A.D. Belegundu, S.D. Rajan, A Shape Optimization Approach Based on Natural Design Variables and Shape Functions, Comput. Methods Appl. Mech. & Eng., Vol. 66, pp 87-106, 1988.

[3] S.D. Rajan, A.D. Belegundu, Shape Optimal Design Using Fictitious Loads, AIAA J., Vol. 27(1), pp. 102-107, 1989.

[4] A.D. Belegundu, S.D. Rajan, Shape Optimal Design Using Isoparametric Elements, Proc.

AIAA/ASME/AHS 29th SDM Conf., Blacksburgh, VA, pp. 696, 701, 1988.

[5] T-M. Yao, K. K. Choi, 3-D Shape Optimal Design and Automatic Finite Element Regridding, Inter. J. Numer. Methods. Eng., Vol. 28, pp. 369-384, 1989.

[6] T. Hosokawa, H. Tsukada, Y. Maeda, T. Nakakubo, M. Nakada, Development of Computer Aided Engineering for Piston Design, SAE Paper # 890775, 1989.

[7] R.T. Haftka, R.V. Grandhi, Structural Shape Optimization-A Survey, Comput. Methods Appl. Mech. & Eng., Vol. 57, pp. 91-106, 1986.

[8] Y. Ding, Shape Optimization of Structures: A Literature Survey, Comput. & Struct., Vol. 24(6), pp. 985-1004, 1986.

[9] K.K. Choi, J.L.T. Santos, T.M. Yao, Recent Advances in Design Sensitivity Analysis and Its Use in Structural Design Process, SAE Technical Paper Series 880783, 1988.

[10] S.D. Rajan, A.D. Belegundu, J. Budiman, An Integrated System for Shape Optimal Design, Comput. & Struct., Vol. 30(1/2), pp. 337-346, 1988.

[11] M.H. Imam, Three-Dimensional Shape Optimization, Int. J. Numer. Methods Eng., Vol. 18, pp. 661-673, 1982.

[12] M.E Botkin, Shape Optimization of Plate and Shell Structures, AIAA J., Vol. 20(2), pp. 268-273, 1982.

[13] V. Braibant, C. Fluery, Shape Optimal Design Using B-Splines, Comput. Methods Appl. Mech, & Eng., Vol. 44, pp 247-267, 1984.

[14] R.J. Yang, M.J. Fiedler, Design Modeling for Large-Scale Three Dimensional Shape Optimization Problems, ASME Computers in Engineering Conference Proceedings, Vol. 2, 1987.

[15] D.A. Tortorelli, Sensitivity Analysis for Nonlinear Constrained Elastostatic Systems, in preparation.

[16] K.J. Bathe, Finite Element Procedures in Engineering Analysis, Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632, 1982.

[17] G.N. Vanderplaats, Numerical Optimization Techniques for Engineering Design - With Applications, (McGraw-Hill, New York, 1984).

[18] D.G. Phelan, C. Vidal, R.B. Haber, Explicit Sensitivity Analysis of Nonlinear Elastic Systems, Computer Aided Optimum Design: Recent Advances, Brebbia & Hernandez, Eds., Comp. Mechanics Publications/Springer-Verlag, pp. 357-366, 1989.

[19] R.B. Haber, A New Variational Approach to Structural Shape Design Sensitivity Analysis, NATO ASI Series, Vol. F27, Computer Aided Optimal Design: Structural and Mechanical Systems. Edited by C.A. Mota Soares, Springer-Veral Berlin Heildelberg 1987.

[20] D.F. Rogers, J.A. Adams, Mathematical Elements for Computer Graphics, McGraw-Hill, Inc., New York, 1976.

[21] S. Kodiyalam, G.N. Vanderplaats, Shape Optimization of 3D Continuum Structures Via Force Approximation Techniques, Proc. AIAA/ASME/AHS 29th SDM Conf., Blacksburgh, VA, pp. 1696-1704, 1988.

[22] E.J. Haug, K.K. Choi, V. Komkov, Design Sensitivity Analysis of Structural Systems, Academic Press, New York, 1986.

[23] B. Barthelemy, R.T. Haftka, Accuracy of the Semi-Analytical Method for Shape Sensitivity Calculations, Proc. AIAA/ASME/AHS 29th SDM Conf., Blacksburgh, VA, pp. 572-588, 1988.

[24] R.J. Yang, A Hybrid Approach for Shape Optimization, ASME Computers in Engineering Conference Proceedings, Vol. 3, 1988.

[25] K. Lee, Shape Optimization of Assemblies Using Geometric Properties, Ph.D. Thesis, Massachusetts Institute of Technology, 1983.

[26] R.J. Yang, A 3-D Shape Optimization Program - SHOP3D, Comput. & Struct., Vol. 31, No. 6, pp. 881-890, 1989.

OPTIMIZATION OF ROTOR BLADES FOR COMBINED STRUCTURAL, DYNAMIC, AND AERODYNAMIC PROPERTIES

by

CHENGJIAN HE
Post-Doctoral Fellow

and

DAVID A. PETERS
Professor

School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332

Presented at the Third Air Force/NASA Symposium on Recent
Advances in Multidisciplinary Analysis and Optimization

San Francisco, California
24-26 September 1990

1 Introduction

Optimal helicopter blade design with computer-based mathematical programming has received more and more attention in recent years. Most of the research has focused on optimum dynamic characteristics of rotor blades to reduce vehicle vibration, Refs. 1 to 4. There is also work on optimization of aerodynamic performance, Ref. 5, and on composite structural design, Ref. 6. This research has greatly increased our understanding of helicopter optimum design in each of these aspects.

Helicopter design is an inherently multidisciplinary process involving strong interactions among various disciplines which can appropriately include aerodynamics, dynamics (both flight dynamics and structural dynamics), aeroelasticity (vibrations and stability), and even acoustics. Therefore, the helicopter design process must satisfy manifold requirments related to the aforementioned diverse disciplines. In our present work, we attempt to combine several of these important effects in a unified manner. First, we design a blade with optimum aerodynamic performance by proper layout of blade planform and spanwise twist. Second, the blade is designed to have natural frequencies that are placed away from integer multiples of the rotor speed for a good dynamic characteristics. Third, the structure is made as light as possible with sufficient rotational inertia to allow for autorotational landing, with safe stress margins and flight fatigue life at each cross-section, and with aeroelastical stability and low vibrations. Finally, a unified optimization refines the solution.

2 Mathematical Formulation

For an optimal design process of a helicopter, our mathematical formulation can be posed as a constrained minimization problem.

The baseline configuration used for the rotor blade is a box-beam model, Ref. 2. The primitive design variables for the box beam are its width, its flange thickness, and its web thicknesses. Two additional primitive design parameters allow for further freedom in weight distribution. These seven design parameters (along with given material properties) define the blade mass and structural properties. Naturally, they are constrained such that the pieces must fit within the aerodynamic envelope.

The first phase in our multi-disciplinary design of rotor blades is to improve the aerodynamic performance. The objective function is chosen to be the power required for the main rotor in hovering flight. In the aerodynamic performance analysis, the blade chord and twist are chosen as design variables. In the current study, we do not confine ourself to linear twist or taper only. A more general distribution is assumed. The chord and twist are expressed in the following polynomial form

$$C(\bar{r}) = c_0 + c_1(\bar{r} - 0.75) + c_2(\bar{r} - 0.75)^2 + \cdots \tag{1}$$

$$\theta(\bar{r}) = \theta_0 + \theta_1(\bar{r} - 0.75) + \theta_2(\bar{r} - 0.75)^2 + \cdots \tag{2}$$

By doing this, the distribution coefficients c_0, c_1, $\cdots$, and θ_0, θ_1, and etc. become design variables. Therefore, we create an opportuity to test various types of possible chord and twist distributions for an optimum aerodynamic design.

For the second phase of optimization, we work on blade natural frequency placement. A finite element approach with coupled blade flapping, lead-lag, and torsion motions is applied as an eigenvalue analysis. In the current study, our goal is to keep the frequencies away from resonance with rotor speed harmonics up to the ninth mode. The prescribed windows are set up such that each of these frequencies is away from any integer multiple of rotor rotational speed. Because stress and flutter are important aspects of the current optimization process, the yield stresses and fatigue life, as well as classical bending-torsion flutter of the blade have been added as constraints in the optimization.

The CONMIN optimization code has been used as optimizer. It is a first-order programing method that uses a feasible search direction obtained from a compromise between gradients of objective function and imposed constraints. Initially, we tried a unified approach where we chose the power required in hover as the objective function with all contraints (autorotational inertia, blade natural frequencies, stress at each cross section, chordwise c.g., and side constraints on design variables). We wanted to obtain an optimum solution in one single run; but, this approach failed. The optimizer stops after it has been unable to obtain desired improvements in the objective function within a reasonable number of iterations.

Thus, we decided to phase the whole optimization process into several stages. First, we use power required in hover as the objective function with autorotation inertia as a constraint. Therefore, the blade chord and twist distributions, together with the lumped mass, are applied as design variables. After the performance optimization, the blade chord and twist have changed significantly, thus resulting in a significant shift in blade natural frequencies. It is then necessary to perform blade frequency-placement optimization to bring them back into the prescribed windows. Next, we turn to minimize the rotor power with all constraints (frequencies, stress, chordwise c.g.). Because blade weight is a strong contributer to power required (for a given payload) and because the aerodynamic taper and twist are already near optimum, this last phase is equivalent to minimizing weight. Thus, at the end, we reach our overall objective (i.e., an optimum aerodynamic performance, properly placed blade natural frequencyies, satisfied blade sectional critical stresses, and free of aeroelastic instabilities).

3   Results and Discussion

The results obtained thus far show the interactions among aerodyanmic performance, blade dynamics properties, and cross sectional stresses. Figures 1a-1d present hover performance optimization results. As shown, the mathematical optimizer, like a helicopter design expert, knows exactly how to make a tapered blade to reduce power required while maintaining thrust needed to lift the vehicle. The original blade has a -8° twist which offers quite good performance (Figure of Merit is 0.76). However, the optimization process has been able to gain more benefit through taper. A 4% saving in total power required and a Figure of Merit of 0.7892 are achieved from the optimum design even though the original design, as mentioned, had a high Figure of Merit. Figure 1c shows that the inboard section has smooth blade loading. Thus, as expected, the power required varies in its distribution, with less power consumed at the tip region (which is critical for the reduction of power

consumption while maintaining the specified lifting capacity), Fig. 1d. These figures also compare the linear, parabolic, and cubic chord and twist designs. It is noted that we obtain a curve-shape blade planform for both parabolic and cubic designs. The chord is not decreased all the way to the tip. Instead, we have a cup-shape there. It is interesting to know that we achieve almost the same final Figure of Merit (close to 0.79) for the design with either linear or higher-order chord and twist variations This is to say that the linear twist and tapered blade produces the essential gain for aerodynamic performance. At least, this conclusion is true for hovering flight. It remains to be tested in forward flight. The higher-order distributions do have a faster convergence rate in the numerical design process than the linear case. Since rotor performance is very sensitive to the blade tip loading, it is interesting to examine different tip loss formulas. Fig. 2 compares the optimized results with Prandtl, Wheatley, Wald, and Peters tip loss approaches. Prandtl's and Peters' formulas yield similar design; Wheatley's formula produces more taper at the tip; and Wald's formula gives a relatively poor design.

Strong interactions exist between aerodynamic performance optimization and the tailoring of blade structural dynamics since the changes in both chord and twist distributions significantly affect blade natural frequencies and mode shapes. Table 1 compares the blade natural frequencies before and after aerodynamics optimization. Large frequency shifts occur on all higher frequency modes. For example, nondimensional frequencies of $1st$ torsion, and $4th$ and $5th$ flapping modes change by more than 15%, and even shift into another frequency window. To accomplish optimum design with respect to both aerodynamics and dynamics, we need to bring these shifted frequencies into prescribed windows. The results from the two-step frequency placement are presented in Table 1. As shown, all frequencies, except the $5th$ flapping mode, are successfully brought into the windows, even though they have been shifted greatly. The $5th$ flapping frequency is slightly off the window, but it is going in the direction toward the inside of the window at each iteration. It just converges slowly.

Both yield and fatigue stresses at each blade cross-section have been imposed as constraints to insure no structural failure due to excessive stresses for final optimum design. Figures 3a and 3b show static and critical stress distributions for the initial blade structure, the optimal aerodynamic performance design, and the final multi-displinary design with stress constraints. The optimal aerodynamic performance design (dashed line) has a significant impact on blade cross-section stresses. In particular, large stresses occur around the critical stress region (55% to 87% radial stations) as compared to the initial design (doted line). Optimization with stress constraints (solid line) improves the design around this critical region. Figure 3c compares the total critical stress distribution for the optimum design with and without stress constraints. The optimum design without stress constraint has a poor fatigue life, with possible structural fatigue failure (at about 82% radial station); but the optimum design with proper stress constraints has lowered the total blade cross-section stresses, offering a safe fatigue life for the optimally designed blades.

To summarize, the optimal aerodynamic performance in hover and proper rotor blade natural frequencies have been achieved with constraints on autorotational inertia, blade chordwise c.g. and blade sectional critical stresses. This has been done by a proper division

of the design optimization into stages. The paper presents various types of taper and twist for optimal aerodynamic performance and shows significant influence of the optimum aerodynamic design on the blade dynamic properties and cross-sectional stresses.

4 References

1. Friedmann, P.P. and Shanthakumaran, P., "Optimum Design of Rotor Blades for Vibration Reduction," *Journal of the American Helicopter Society*, Vol. 29(4), Oct. 1984, pp.70-80.

2. Peters, David A., Rossow, Mark P., Korn, Alfred, and Ko, Timothy, "Design of Helicopter Rotor Blades for Optimun Dynamic Characteristics," *Computers and Mathematics with Applications*, Vol. 12(1), 1986, pp. 85-109.

3. Lim, Joon W. and Chopra, Inderjit, "Aeroelastic Optimization of a Helicopter Rotor," Proceedings of the 44th Annual National Forum of the American Helicopter Society, Washington, DC, June 16-18, 1988.

4. Davis, M.W. and Weller, W.H., "Application of Design Optimization Techniques to Rotor Dynamics Problems," *Journal of the American Helicoptr Society* , Vol. 33(3), July 1988.

5. Walsh, Joanne L., Bingham, Gene J., and Riley, Michael F., "Optimization Methods Applied to the Aerodynamic Design of Hellicopter Rotor Blades," *Journal of the American Helicoptr Society* , Vol. 32(4), October 1987, pp. 39-44.

6. Lake, R.C., and Nixon, M.W., "A Preliminary Investigation of Finite-Element Modeling in Analysis for Composite Rotor Blade," Proceedings of the Second International Conference on Rotorcraft Basis Research, College Park, Maryland, February 1988.

Table 1 Frequency Variations

| Items	Original	After Aero. Opt.	Frequency Placement (I)	Frequency Placement (II)	Weight Minimization
inplane-1st	0,4745	0.4683	0.4725	0.4756	0.4756
flapping-1st	1.0278	1.0236	1.0247	1.0252	1.0252
flapping-2nd	2.7509	2.7740	2.7323	2.6860	2.6840
torsion-1st	4.1327	5.2148	4.8270	4.7000	4.7000
flapping-3rd	4.9585	4.6968	4.5632	4.5474	4.5439
inplane-2nd	6.7869	6.0735	5.8591	5.6382	5.6458
flapping-4th	8.1068	6.9694	6.7932	6.6643	6.6635
flapping-5th	12.1980	10.4555	10.0148	9.7735	9.7499
torsion-2nd	12.6973	12.8909	12.8919	13.4074	13.3044
blade weight	201	199	212	222	221
Autorotation	12453	12581	13609	14516	14432
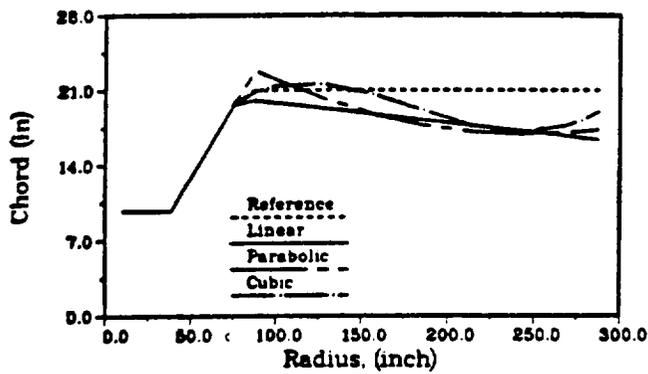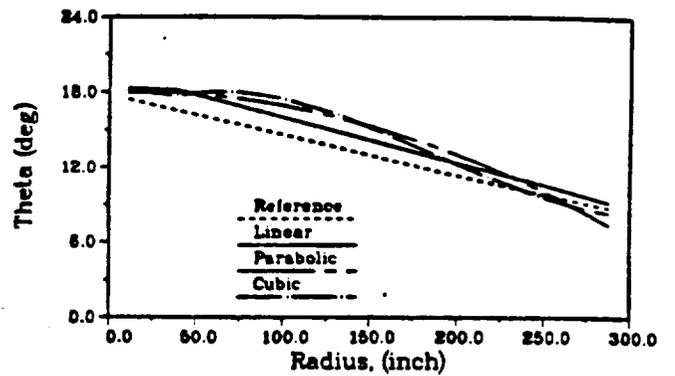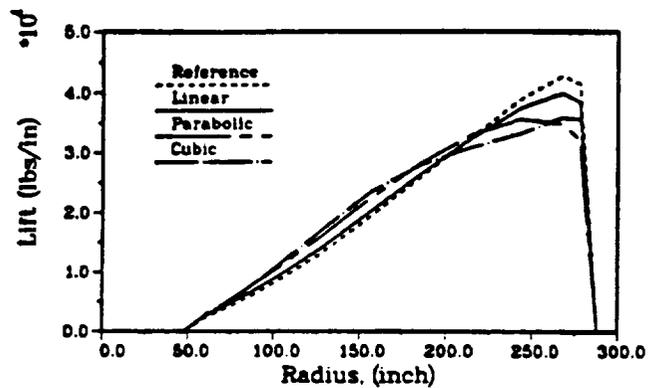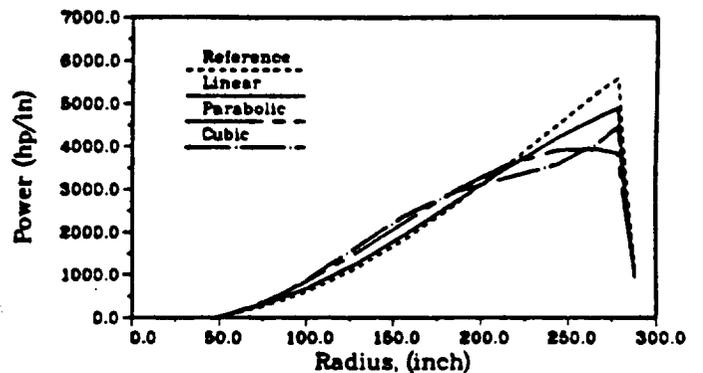
Figure 1   Comparison of Linear, Parabolic and Cubic
Chord and Twist Designs, (Continued)
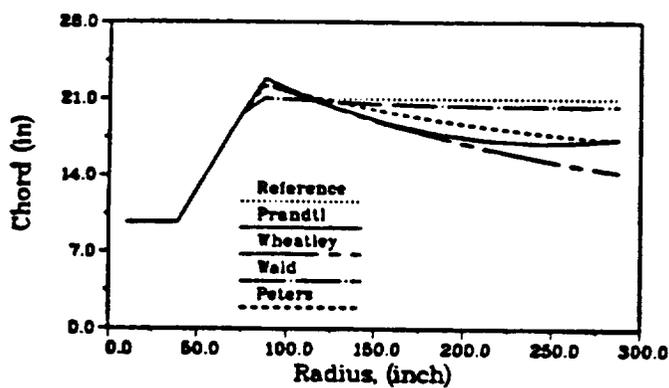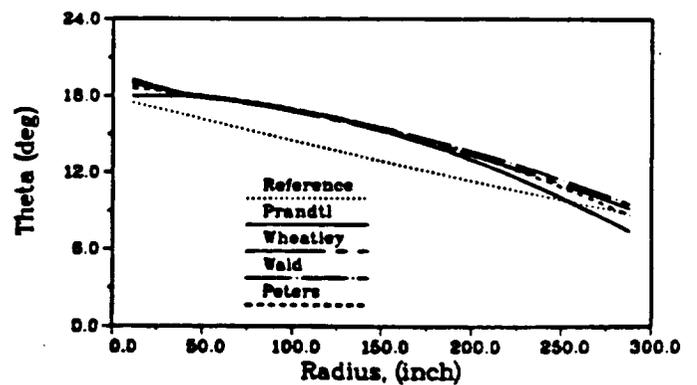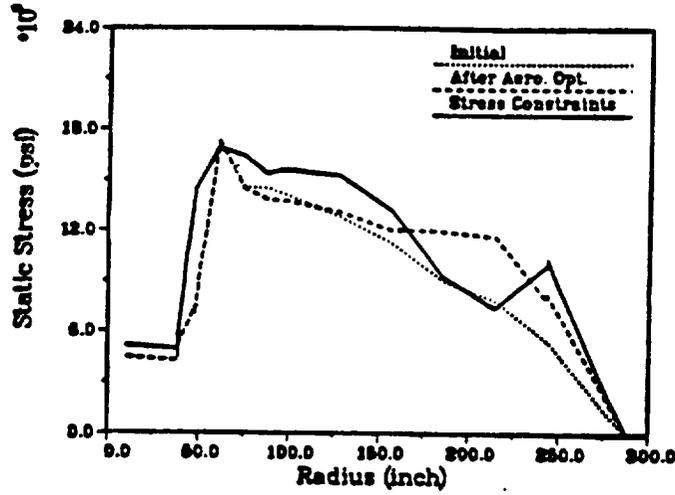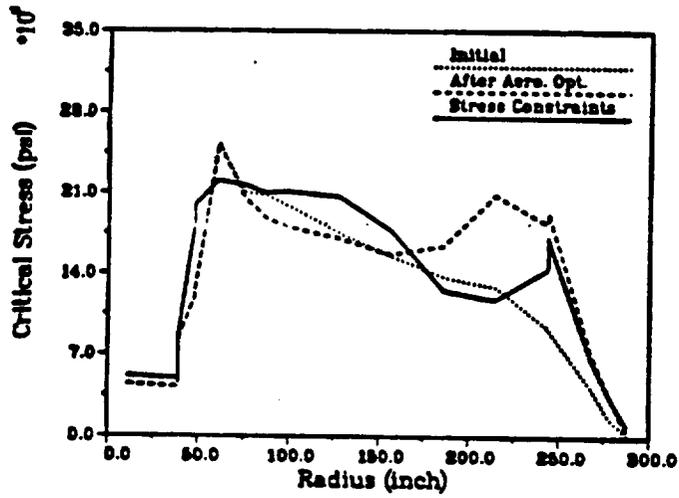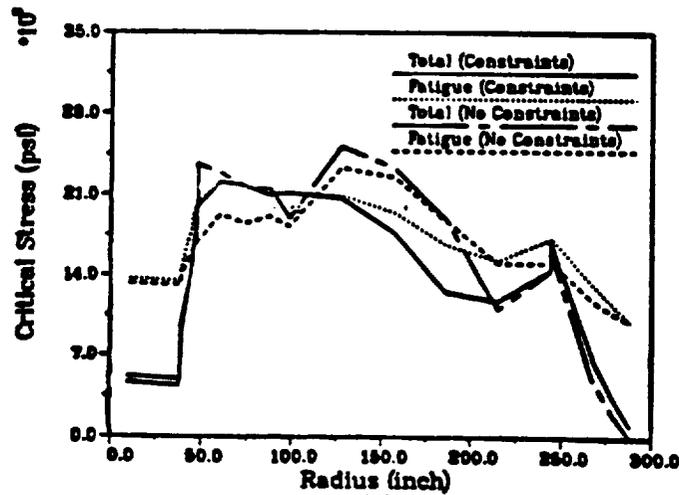

Figure 2   Comparison of Different Blade Tip Loss
Formulations.

(a)

(b)

(c)

Figure 3   Comparison of Stresses with and without Stress Constraints

240

RANDOM SEARCH OPTIMIZATION
BASED ON
GENETIC ALGORITHM AND DISCRIMINANT FUNCTION

by

M.O. Kıcıman[1], M. Akgül[2], G. Erarslanoğlu[3]

ABSTRACT

The general problem of optimization with arbitrary merit and constraint functions, which could be convex, concave, monotonic or non–monotonic, is treated using stochastic methods. To improve the efficiency of the random search methods, a genetic algorithm for the search phase and a discriminant function for the constraint–control phase were utilized. The validity of the technique is demonstrated by comparing the results to published test problem results. Numerical experimentation indicated that for cases where a quick near optimum solution is desired, a general, user–friendly optimization code can be developed without serious penalties in both total computer time and accuracy.

INTRODUCTION

Several commercially available optimization programs with particular application to aircraft structures and performance have been in use for decades. A great majority of these programs are deterministic and based on mathematical nonlinear programming algorithms. They also require the evaluation of at least the first derivative of several functions involved in the calculation, user decision for the step lengths and, monotonic functions for best results. Several methods have also been tried to avoid the local optimum versus the global optimum handicap. On the other hand the most important argument against the stochastic search procedures is their inefficiency and impracticality for meaningful real world problems involving very large number of design variables and constraint functions. With the new computer capabilities, both the memory requirements and the execution time lengths for even the most complex optimization problems ceased to be decisive factors for the chioce of solution methods. Thus the total engineering time required for a feasible optimum solution became the most important parameter. Random search methods are inherently easy to use and do not require special engineering expertise, therefore improvement efforts were directed mainly to the savings in the computer CPU+RESOURCE times. Replacement of the total unbaised random search with more logical improvement methods such as the genetic algorithm method dates back to the Fogel's original dissertation titled " Artificial Intelligence Through Simulated Evolution "[1] and since then it has been subjected to several studies [2,3,4].

As of this time the authors are not aware of the existence of a complete mathematical proof of convergence for the genetic algorithm methods, only numerical results showing convergence are available, and they are promising. The other important cost saving area is in the time spent on checking the design points against the constraint functions to decide whether they are acceptable or not. A method based on the discriminant function idea has been utilized to reduce the effort of checking against the actual constraints by replacing them with a hyper separation surface [5,6]. By minimizing the risk of accepting an unacceptable design point or rejecting an acceptable point this quasi–constraint technique provided time saving solutions.

DEVELOPMENT OF THE SEARCH PROCEDURE

The procedure has four phases:
1. Initial grid sampling and evaluation.
2. Establishment of the Discriminant function, which is to be utilized as a Quasi–constraint function.
3. Generation of new design points via genetic algorithm, and evaluation with the discriminant function.
4. Control of final near–optimum design points with actual constraints.

1. Initial Grid Sampling

In an optimization problem involving n variables the design point $X(x_1, x_2, \ldots, x_n)$ is the member of the design

---

[1] Director, R& D and Engineering, Turkish Aerospace Industries, TAI–Mürted, Turkey.
[2] Associate Professor, Bilkent University, Bilkent–Ankara, Turkey.
[3] Senior Specialist, Turkish Aerospace Industries, TAI–Mürted, Turkey.

C-4

set S, which contains all the possible combinations of the design parameters. S can also be considered as an n dimensional volume in a design space bounded by the side constraints. This hyper volume is divided into two regions called the region of acceptable design points A and the region of unacceptable design points U. Both A and U are assumed to be non–multiply–connected regions. In other words all the main constraints can be combined into a single hypersurface dividing the two regions.

By partioning each design parameter into t parts, t^n cells can be formed in S and, assigning one X to the center of each cell initial grid sampling can be performed. This type of a uniformly distributed sampling provides information in an unbaised manner about the whole space.

2. Establishment of the Discriminant Function

The initial sampling provides a set of points in the region of acceptable points and another set in the region of unacceptable design points. Since both regions are not multiply connected these two sets are disjoints and a criterion can be obtained to distinguish between the elements of these two sets. By using linear programming techniques a hyper surface can be found that separates the region of acceptable points from the region of unacceptable design points. This hypersurface then can be used to replace the actual main constraint hyper surface for checking new design points. The main idea here is the utilization of the information obtained during the grid sampling phase to create an artificial constraint surface which is much easier to check against than the actual constraints. Mathematical derivations involved in the discrimination technique are given in Appendix I and reference [7].

3. Genetic Algorithm

In this phase acceptable design points with best merit function attributes are selected as the parent population for the second generation design points. In most genetic algorithm applications design variables are represented in strings of zeros and ones in a binary manner e.g. $(0,0,1,1,0,1,0)$. However in this study actual values of the design parameters are interchanged and mutated randomly to produce the characteristics of the new generation of design points.

A random number generator is used to decide which parameter of which parent will show up in the next generation and what will be the magnitude of the change in that parameter if any. In other words an attempt to mimick the randomness of the natural selection procedure has been made. This way each new generation had a chance to inherit the best characteristic of their parents, who were already members of the top performance set, but not necessarily in the exact magnitude.

Selection procedure was repeated until no appreciable improvement in the MF can be obtained. In this phase, decision of acceptability is based only on the discriminant function filtering.

4. Final Control

At the end of the genetic search, final control of the selected near optimum points were performed against all the actual constraints to ensure the acceptability of the solution. More than one, acceptable and near–optimum design points provided so that the designer will have several alternatives to choose from with similar optimality but different design parameter values.

CONCLUDING REMARKS

The aim of this study was to illustrate the feasibility of a direct, non–gradient–dependent search method which also possesses good economic characteristics such as total engineering and computational expenses. This was achieved by short cutting the time consuming constraint checks with the use of discriminant function and, move–direction and step–size computations with the use of genetic search. Obviously more numerical experimantation both with test problems and, meaningful real life problems are required to improve the present crude attempt.

Appendix I: Linear and Quadratic Discrimination

Given $A = \{a_1, a_2, \ldots, a_k\}$ of set of acceptable points and $U = \{u_1, u_2, \ldots, u_\ell\}$ of unaccepatable points in $\mathcal{R}^n$, we would like to determine a *hyperplane* $cx = \alpha$ *separating* A *and* U if possible. If the vector $c \in \mathcal{R}^n$, $\alpha \in \mathcal{R}$ satisfies the conditions:

$$a_i x \leq \alpha \quad i = 1, \ldots, k \tag{1}$$

$$u_j x \geq \alpha \quad j = 1, \ldots, \ell \tag{2}$$

then we say the hyperplane $cx = \alpha$ separates A and U. Assuming such a separation is possible, one can find such an hyperplane by solving the linear program (LP):

$$\max_{c,\alpha,\delta} \delta \qquad (3)$$

LP$_1$: \qquad subject to

$$a_i c \leq \alpha - \delta \quad i = 1, \ldots, k \qquad (4)$$

$$u_j c \geq \alpha + \delta \quad j = 1, \ldots, \ell \qquad (5)$$

This LP is always feasible, $(0,0,0)$ is feasible and for any feasible (c, α, δ) and $\beta > 0$, $\beta(c, \alpha, \delta)$ is again feasible. So one needs to add a *normalization* or *boundedness* conditions to guarentee that the above LP has a finite optimum. One such possibility is as follows:

$$||c||_\infty = 1 \qquad \text{or} \qquad -e \leq c \leq e \qquad (6)$$

where e is a vector of 1's of suitable dimension. Now we can recast LP$_1$ with (6) as

$$\max_{c,\,\alpha,\,\delta} \delta$$

LP$_2$

$$\text{subject to:} \qquad (4\text{-}6)$$

This LP is always feasible and bounded. Let

$$\Delta = 2\delta = \inf\{cu_j \ : \ j = 1, \ldots, \ell\} - \sup\{ca_i \ : \ i+1, \ldots, k\}$$

If $\Delta \geq 0$ ($\Delta < 0$) then a linear separation is possible (impossible). When When $\delta < 0$, if we delete points satisfying

$$\alpha + \delta \leq cx \leq \alpha - \delta$$

then the remaining points are separable. Thus LP$_2$ will find the 'best' separation in the above sense.

Weighted Linear Separation

Suppose, furher, we assign weights w_i to a_i and w_i' to u_j with conditions

$$w_i \geq 0, \qquad w_j' \geq 0, \qquad \text{and}, \qquad W \equiv \sum_i w_i = \sum_j w_j' = 1 \qquad (7)$$

By introducing slack variables s_i and z_j to (4) and (5) we rewrite them as

$$a_i c - \alpha + \delta + s_i = 0 \quad i = 1, \ldots, k \qquad (8)$$

$$u_j c - \alpha - \delta - z_j = 0 \quad j = 1, \ldots, \ell \qquad (9)$$

$$s \geq 0, \quad z \geq 0 \qquad (10)$$

Because of (7) we have the equation

$$\sum_i w_i(\delta + s_i) + \sum_j w_j'(\delta + z_j) = \delta + \sum w_i s_i + \sum w_j' z_j. \qquad (11)$$

We can an LP to find the best weighted linear separation as

$$\max_{c,\,s,\,z,\,\alpha,\,\delta} \quad 2\delta - \sum w_i s_i - \sum w_j' z_j$$

LP$_3$

$$\text{s.t.} \quad (6),\ (8-10)$$

Clearly LP$_3$ is feasible and bounded and for any optimal solution $\Delta = 2\delta$ is the maximum weighted linear separation. As before $\Delta \geq 0$ means separation is possible, othwerwise it is not possible.

Quadratic Separation

Now we would like to separate the sets A and U with quadratic function

$$f(x) = x^T D x + c^T x - \alpha. \tag{12}$$

In other words, we would like to determine $D \in \mathcal{R}^{n \times n}$, $c \in \mathcal{R}^n$, $\alpha \in \mathcal{R}$ so that $f(a_i) \leq 0$, $\forall i$ and $f(u_j) \geq 0$, $\forall j$. Since the quadratic term $x^T D x$ is linear in D, we can cast the 'best' quadratic separation problem with normalization condition similar to (6), in the spirit of LP$_2$, as:

$$
\begin{array}{lll}
& \max \quad \delta & (13) \\
\text{LP}_4 & a_i^T D a_i + a_i^T c \leq \alpha - \delta \quad \forall\, i & (14) \\
& u_j^T D u_j + u_j^T c \geq \alpha + \delta \quad \forall\, j & (15) \\
& -1 \leq d_{ij} \leq 1 \qquad \forall\, i,\, j & (16) \\
& -1 \leq c_i \leq 1 \qquad \forall\, i & (17)
\end{array}
$$

Clarly LP$_4$ is feasible, simply because $D = 0$, $c = 0$, $\alpha = \delta = 0$ is a feasible solution, and it is bounded because of (16) and (17). Hence if $\delta > 0$ or $\delta = 0$ and ($D \neq 0$ or $c \neq 0$), separation is possible. If separation is not possible, we can reformulate the LP so that for the new LP will have $\delta < 0$. A possible method is setting

$$d_{ij} = \pm 1 \quad \text{or} \quad c_i = \pm 1 \tag{18}$$

for some (i, j) or i respectively.

Clearly a weighted version of LP$_4$ can be written similar to LP$_3$.

Enclosing the A region by hyperplanes

Now consider LP$_1$ with normalization condition (6) and assume that linear separation is not possible. Then $c = 0$, $\alpha = \delta = 0$ is an optimal solution. Then one can find a 'best' separation by requiring, as in LP$_4$

$$c_i = 1 \quad \text{or} \quad c_i = -1$$

for some i. To fix the notation let us rewrite the resulting LP as:

$$
\begin{array}{lll}
& \max \quad \delta & \\
\text{LP}'(i_o, +) & & \\
& a_i c \leq \alpha - \delta \quad \forall i & \\
& u_j c \geq \alpha + \delta \quad \forall j & \\
& -1 \leq c_i \leq 1 \quad \forall i & (19) \\
& c_{i_\bullet} = 1 &
\end{array}
$$

Let the resulting half space $H_{i_\bullet}^+ = \{x \, : \, c^* x \leq \alpha^* \}$, where $(c^*, \alpha^*, \delta^*)$ is an optimal solution of (19). Clearly one can define LP$'(i_o, -)$, and $H_{i_\bullet}^-$ similarly. Then

$$\left(\cap_{i=1}^n H_i^+ \right) \cap \left(\cap_{j=1}^n H_j^- \right)$$

can be considered as the acceptable region found by the separation mechanism. In other words, we find at most $2n$ hyperplanes of the form $cx \leq \alpha$ which contains acceptable region. It should be noted that one can relax $c^* x \leq \alpha^*$ condition to $c^* x \leq \alpha^* - \delta^*$ condition to obtain a bigger estimate of the acceptable region.

APPENDIX II – Results of Numerical Experimentations

Two sample problems were treated to illustrate the workings of the optimization procedure. The first one is a well known control problem credited to Rosen and Suzuki [5] and the second one is chosen to show the application of the method to a real aeroelastic tailoring problem.

The following standard notation will be used:

$MF \stackrel{\mathrm{df}}{=}$ Design Objective function to be minimized

$X \stackrel{\mathrm{df}}{=}$ Design vector in an n dimensional space, where n is the number of variables

$G_j \stackrel{\mathrm{df}}{=}$ Constraint functions, for both side constraints and main constraints.

Test Problem # 1

$X = (x_1, x_2, x_3, x_4)$

$MF = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4$

With side constraints $-3 \leq x_i \leq +3$ $i = 1, 2, 3, 4$ and with main constraints:

$G_5)$ $-x_1^2 - x_2^2 - x_3^2 - x_4^2 - x_1 + x_2 - x_3 + x_4 + 8 \geq 0$

$G_6)$ $-x_1^2 - 2x_2^2 - x_3^2 - 2x_4^2 + x_1 + x_4 + 10 \geq 0$

$G_7)$ $-2x_1^2 - x_2^2 - x_3^2 - 2x_1 + x_2 + x_4 + 5 \geq 0$

    Grid Sampling Consisted of 5^4, a total of 625 design points. (53) acceptable, satisfying all the constraints and, (572) unacceptable, failing in at least one constraint, points were located. The best MF value for an acceptable point, obtained from the grid sampling was (-33.84). The discriminant function based on the grid sampling results was obtained using a linear programming (LP) type solution described in Appendix I is:

$-1.2 \leq x_1 \leq 1.2, \quad -1.2 \leq x_2 \leq 1.2, \quad -2.4 \leq x_3 \leq 2.2, \quad -1.2 \leq x_4 \leq 2.4$ for linear DF and,

$x^T D x \leq 0 \Rightarrow x \in A ; \quad x^T D x > 0 \Rightarrow x \in U$ for quadratic DF

where

$$D = \begin{pmatrix} 0 & 1 & 1 & 1 \\ -1 & 0 & 1 & 1 \\ -1 & -1 & 0 & 1 \\ -1 & -1 & -1 & 0 \end{pmatrix}$$

    Genetic Algorithm search consisted of selecting the four acceptable points with best merit functions and cross breeding them with a random number generator that picks both the parent and the amount of variation for the particular design variable. In this problem max variation was limited to 40 percent. Genetic search results are shown in figure 1. At the end fourth generation search, optimum design point was given as: $X = (0.000, 0.864, 2.016, -0.576)$ with $MF = -41.48$ which corresponds to the best point in the second generation meaning that further cross breeding did not improve the solution. True optimum point was $X = (0.000, 1.000, 2.000, -1.000)$ with $MF = -44.00$.

Test Problem # 2

    In this problem the design variables are the orientation angles of the plies in a laminated composite box beam. The box beam represents the main torque box of a swept-back wing [8]. There are several side constraints and main constraints such as the minimum and maximum number of plies, minimum and maximum separation of ply angles, plus strength, flutter and divergence requirements; but the main constraint functions are reduced to just one constraint involving the required deflection shape of the wing under the given load conditions. The merit function is the difference between the required shape of twist and the actual shape of twist obtained for the given design point (Figure 2).

$X = (x_1, x_2, x_3, x_4) \quad\quad x_i = \theta_i \quad\quad i = 1, 2, 3, 4 \ (\ x_4 \text{ is taken as } x_1 + 90)$

$$MF = \sum_{j=1}^{6} | \Phi_j(\text{required}) - \Phi_j(\text{obtained}) | \quad\quad \text{(for a wing box with six sections)}$$

Main constraint function G is $| (\delta_j(\text{given}) - \delta_j(\text{obtained}))/\delta_j(\text{given}) | \leq 0.1 \ \forall j$

    The main idea for this aeroelastic tailoring optimization problem is to obtain a twist shape as close as possible to the ideal twist shape under a given set of load conditions while satisfying a given deflection shape in addition to the usual strength and stiffness requirements. The results are shown in Figure 2.

    The discriminant function obtained with the utilization of the grid sampling results is given below:

$-18.8 \leq x_1 \leq 3.2, \quad -52.1 \leq x_2 \leq -41.1, \quad -128.9 \leq x_3 \leq -111.1, \quad -71.2 \leq x_4 \leq 93.2$

for linear DF and

$$x^T D x \leq 0 \Rightarrow x \in A ; \quad x^T D x > 0 \Rightarrow x \in U$$

for quadratic DF where

$$D = \begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & 1 \\ 1 & -1 & 0 \end{pmatrix}$$

REFERENCES

[1.] Fogel, L.J., Owens, A.J. and Walsh, M.J., " *Artificial Intelligence Through Simulated Evolution* ", John Wiley & Sons, Inc., New York, N.Y. 1966.

[2.] Grefenstette, J., ed., " *Proceedings of the International Conference on Genetic Algorithms and Their Applications* ", Pittsburgh, PA, 1985.

[3.] Hajela, P.," *Genetic Search - an Approach to the Nonconvex Optimization Problem* ", Proceedings of the 30th. AIAA/ASME/AHS Structures, Structural Dynamics and Materials Conference, Mobile Alabama, April, 1989, pp. 165–175.

[4.] Bock, K.W., " *Aerodynamic Design by Optimization* ". AGARD–CP–463 Computational Methods for Aerodynamic Design (inverse) and Optimization, Loen, Norway, May 1989, pp. 20-1-20-12.

[5.] Kıcıman, M.O., Akgül, M. " *Application of Discriminant Function Technique to Random Search* ". AGARD Conference Proceedings No. 123, Second Symposium on Structural Optimization, 1973.

[6.] Kıcıman, M.O., " *Application of Discriminant Function Technique to Structural Optimization* ", Turkish Scientific and Technical Research Council Rep. MAG–327/A, 1974.

[7.] Mangasarian, O.L., " *Linear and Nonlinear Separation of Patterns by Linear Programming* ", Operations Research 13 (1965) 444–452.

[8.] Kıcıman, M.O.," *Strength–Twist Aeroelastic Tailoring of a Torque Box* ", AC–88, (TFD–76–708) Rockwell International LADD, Sept. 1976.
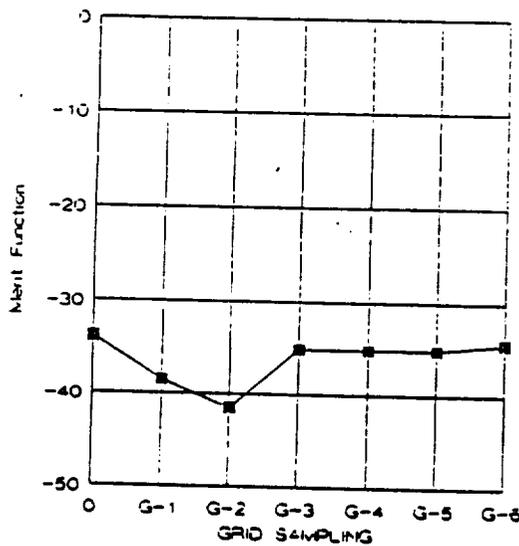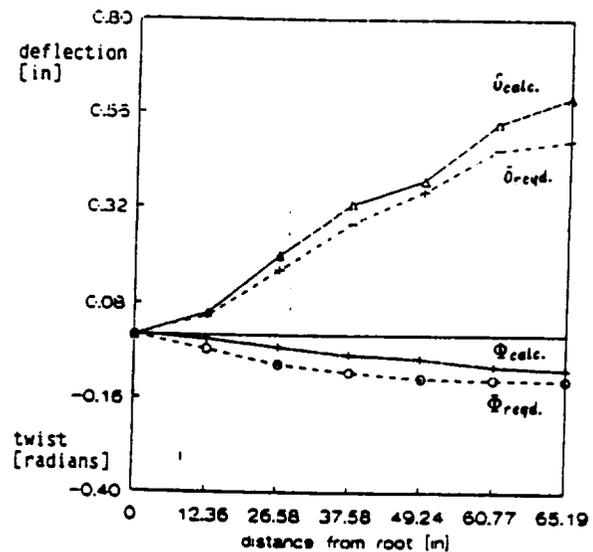
Figure 1. Test Problem 1 Results.

Figure 2. Test Problem 2
Deflection and Twist

Newton Modified Barrier Method In Constrained Optimization

R. Polyak

Mathematical Sciences Department

IBM Research Division

T. J. Watson Research Center

Yorktown Heights, NY 10598

1. <u>Introduction.</u> In this paper we developed and investigated the Newton Method for solving constrained (non-smooth) optimization problems. This approach is based on the Modified Barrier Functions (MBF) theory (see [4]) and on the global converging stepsize version of the Newton Method for smooth unconstrained optimization (see for example [1]). Due to the excellent properties of the MBF near primal-dual solution, the Newton Modified Barrier Method (NMBM) has a better rate of convergence, better complexity bound and is much more stable in the final stage of the computational process than the methods, which are based on the Classical Barrier Functions (CBF) (see [6] and bibliography).

2. <u>Modified Barrier Function.</u> Let $f_0(x)$ and $-f_i(x)$, $i = \overline{1, m}$ be convex, $f_i(x) \in C^2$, $i = 0, \overline{m} : R^n \to R^1$, and there exists

$$(1) \qquad x^* = \arg\min \{f_0(x)/x \in \Omega\}$$

where $\Omega = \{x : f_i(x) \geq 0, i = 1, \overline{m}\}$ (int $\Omega \neq \emptyset$). Let $\{i : f_i(x^*) = 0\} = \{1, \dots, r\}$, $f(x) = (f_i(x), i = 1, \overline{m})$, $f_{(r)}(x) = (f_i(x), i = 1, \overline{r})$, $f'_i(x) = \operatorname{grad} f_i(x)$, $i = 0, m$, $f'(x) = J(f(x))$, $f'_{(r)}(x) = J(f_{(r)}(x))$ the Jacobi matrix of the vector functions $f(x)$, $f_{(r)}(x)$ respectively. Because int $\Omega \neq \emptyset$ the Karush-Kuhn-Tucker conditions holds true, i.e., there exists a vector $u^* = (u^*_1, \dots, u^*_m) \geq 0$

$$(2) \qquad L'_x(x^*, u^*) = f'_0(x^*) - \sum_{i=1}^{m} u^*_i f'_i(x^*) = 0, \ f_i(x^*) u^*_i = 0, \ i = \overline{1, m}.$$

We also suppose that the standard second order optimality sufficient conditions are fulfilled.

$$(3) \qquad \operatorname{rank} f'_{(r)}(x^*) = r, \ u^*_i > 0, \ i = \overline{1, r}$$

247

(4)
$$(L''_{xx}(x^*, u^*)y, y) \geq \lambda(y, y), \lambda > 0 \; \forall y \neq 0 \; : f'_{(r)}(x^*)y = 0$$

i.e., the problem (1) is nondegenerate. Let $k > 0, \Omega_k = \{k f_i(x) + 1 \geq 0, i = \overline{1, m}\} \supset \Omega = \{x : k^{-1} \ln (k f_i(x) + 1) \geq 0, i = \overline{1, m}\}$. Therefore, the problem

(5)
$$x^* = \arg \min \{f_0(x) \mid k^{-1} \ln (k f_i(x) + 1) \geq 0, i = \overline{1, m}\}$$

is equivalent to (1).

Let us consider the Classical Lagrangian $L(x, u, k) = f_0(x) - k^{-1} \sum_{i=1}^{m} u_i \ln (k f_i(x) + 1)$ for the equivalent problem (5). The Modified Barrier Function, which corresponds to the Frisch Classical Barrier Function $\varphi(x, k) = f_0(x) - k^{-1} \sum_{i=1}^{m} \ln f_i(x)$ [2], we define by formula

$$F(x, u, k) = \begin{cases} L(x, u, k) , & x \in \operatorname{int} \Omega_k \\ \infty & , & x \bar{\in} \operatorname{int} \Omega_k \end{cases} .$$

The MBF properties have been investigated in [4]. Here we are going to mention only the basic facts. Let $0 < \varepsilon < \min \{u^*_i / i = \overline{1, r}\}$, $D_i(\cdot) = D_i(u^*, k_0, \delta, \varepsilon) = \{u_i : u_i \geq \varepsilon, |u_i - u^*_i| \leq \delta k, k \geq k_0 > 0\}, i = \overline{1, r}, \cdot D_i(\cdot) = D_i(u^*, k_0, \delta, \varepsilon) = \{u_i : 0 \leq u_i \leq \delta k, k \geq k_0 > 0\}$, $D(u^*, k_0, \delta, \varepsilon) = D_i(\cdot) \otimes \ldots \otimes D_r(\cdot) \otimes \ldots \otimes D_m(\cdot)$; $U_k = \{u = (u_1, \ldots u_m) \geq 0 : \max \{\varepsilon, u^* - \delta k\} \leq u_i \leq \delta k, 1 \leq i \leq r, 0 \leq u_i \leq \delta k, i = \overline{r + 1, m}\}$, $\Omega^* = \{x \in \Omega : f_0(x) = f_0(x^*)\}$, $\|x\| = \max_{1 \leq i \leq n} |x_i|$.

3 Basic Theorems.

__Theorem 1__ Let $f_0(x)$ and $-f_i(x), i = \overline{1, m}$ are convex and smooth, Ω is compact then for any $u = (u_1, \ldots, u_m) > 0$ and any $k > 0$

1) there exists such a vector $\hat{x} \equiv \hat{x}(u, k) = \arg \min \{F(x, u, k) / x \in R^n\}$, and $\hat{u} = \hat{u}(u, k) = [\operatorname{diag}(k f_i(\hat{x}) + 1]_{i=1}^{m} u$ that $F'_x(\hat{x}, u, k) = L'_x(\hat{x}, \hat{u}) = f'_0(\hat{x}) - \sum_{i=1}^{m} \hat{u}_i f'_i(\hat{x}) = 0$

2) $\hat{x}(u^*, k) = x^* = \operatorname{argmin} \{F(x, u^*, k) \mid x \in R^n\}$ $F(x^*, u^*, k) = f_0(x^*)$ and $F'_x(x^*, u^*, k) = L'_x(x^*, u^*) = f'_0(x^*) - \sum_{i=1}^{m} u^*_i f'_i(x^*) = 0$

3) $\hat{u}(u^*, k) = u^*$, i.e. u^* is a fixed point of the mapping $u \to \hat{u}(u, k)$

__Theorem 2__ Let $f_i(x) \in C^2$, $i = 0, \overline{m}$ and the conditions (2) - (4) are fulfilled then:

248

1) for any $(u, k) \in D(u^*, k_0, \delta, \varepsilon)$ there exists $\hat{x} = \hat{x}(u, k) = \hat{x}(\cdot)$ and $\hat{u} = \hat{u}(u, k) = \hat{u}(\cdot)$ such that $F'_x(\hat{x}, (u, k), u, k) = 0$, the estimation

$$(6) \qquad \max\{\|\hat{x} - x^*\|, \|\hat{u} - u^*\|\} \le ck^{-1}\|u - u^*\| = \gamma_k\|u - u^*\|$$

holds true and c independent of $k \ge k_0$.

2) $F(x, u, k)$ is strongly convex in the neighborhood of $\hat{x} = \hat{x}(u, k)$ uniformly on $(u, k) \in D(u^*k_0, \delta, \varepsilon)$ i.e., there exists $\mu_0 > 0$ that

$$\text{mineigval } F''_{xx}(\hat{x}(u, k), u, k) \ge \mu_0 > 0, \quad \forall (u, k) \in D(u^*, k_0, \delta, \varepsilon).$$

3) Let $U^* = [\text{diag } u^*_i]^r_{i=1}$ then $F''_{xx}(x^*, u^*, k) = L''_{xx}(x^*, u^*) + k f'^T_{(r)}(x^*)U^* f'_{(r)}(x^*)$ and there exists such $\mu > 0$ that

$$(7) \qquad \text{mineigval } F''_{xx}(x^*, u^*, k) \ge \mu > 0, \quad \forall k \ge k_0.$$

4) Let $k \ge k_0$ fixed and $M = \text{maxeigval } F''_{xx}(x^*, u^*, k)$ then there exists $0 < \alpha < 1$, which is independent of $u \in U_k$ that the next inequality

$$(8) \qquad \text{cond } F''_{xx}(\hat{x}(u, k), u, k) \ge \alpha \ \text{cond } F''_{xx}(x^*, u^*, k) \ge \alpha \mu M^{-1}$$

holds uniformly in $u \in U_k$, i.e. cond $F''_{xx}(\hat{x}(u, k), u, k)$ is stable for any fixed $k \ge k_0$.

Remark 1 We would like to emphasize that theorem 2 takes place without the assumption of convexity $f_0(x)$ and $-f_i(x)$, $i = 1, \overline{m}$

4. Newton Modified Barrier Method (NMBM) We consider a bounded set $Y = X \times U : X \subset R^n$, $U \in R^m$, $x^* \in X$, $u^* \in U$ and $k > 0$. On the $Y \times R^1_+$, we consider a nonnegative function $v(y, k) \equiv v(x, u, k) = \max\{-\min_{1 \le i \le m} f_i(x), \|F'_x(x, u, k)\|, \sum_{i=1}^{m} u_i |f_i(x)|\}$. It is clear that for any convex programming problem, the next relation

$$(9) \qquad v(y, k) = 0 \Rightarrow y = y^* = (x^*, u^*)$$

holds true for any $k > 0$. Along with $\Omega_k \supset \Omega$ we consider a set $\Omega_k^+ = \{x : f_i(x) \geq k^{-1}, i = 1, \overline{m}\} \supset \Omega$, a small enough $\varepsilon > 0$ and a monotone increasing sequence $\{k_s\}_{s=1}^\infty : k_{s+1} > k_s, \lim_{s \to \infty} k_s = \infty$. Let $k = k(0) = k_1 > k_0$, $d(0) = 1$, $0 < \gamma \leq \frac{1}{2}$ is fixed.

Now we are going to describe the NMBM. We will start with $x = \overline{x}^0 = x^0 \in \Omega_k^+$, $u^0 = e = (1, \dots, 1) \in R^m$, and let $\overline{x}^s, x^s, u^s, k(s), d(s)$ have already been found. To find the approximation (x^{s+1}, u^{s+1}) one has to fulfill the next operation.

0. Start $x := \overline{x}^s = \overline{x}$.

1. Set $u := u^s, k := k(s), d := d(s)$.

2. Find $\zeta = \zeta(x, u, k)$ by solving the system

(10)
$$F''_{xx}(x, u, k) \zeta = -F'_x(x, u, k)$$

and set $t := 1$.

3. Check $x + t\zeta \in \Omega_k$ and $F(x + t\zeta, u, k) - F(x, u, k) \leq \frac{1}{3} t (F'_x(x, u, k), \zeta)$.

4. If $x + t\zeta \in \Omega_k$, the inequality is fulfilled and $t = 1$ set $x := x + \zeta$ and go to 5, if $x + t\zeta \in \Omega_k$, the inequality is fulfilled and $t < 1$ set $x := x + t\zeta$ and go to 2, if $x + t\zeta \overline{\in} \Omega_k$ or the inequality does not fulfill set $t := \frac{t}{2}$ and go to 3.

5. If $\|\zeta\| \leq \varepsilon$ go to 6, otherwise go to 2.

6. Set $\hat{x} := x, \hat{u} := [\mathrm{diag}\,(k f_i(\hat{x}) + 1)^{-1}]_{i=1}^m u, \hat{y} = (\hat{x}, \hat{u})$ and check $v(\hat{y}, k) \leq \varepsilon$. If $v(\hat{y}, k) \leq \varepsilon$ set $y^* := \hat{y}$ otherwise, i.e. if $\varepsilon < v(\hat{y}, k) \leq \gamma^{d+1}$ set $x^{s+1} := \hat{x}, u^{s+1} := \hat{u}$, start $x := x^{s+1}, d(s+1) := d(s) + 1$, $k(s+1) := k(s), s + 1 := s$ and go to 1.

7. If $v(\hat{y}, k) > \gamma^{d+1}$ set $\hat{t} = \max\{t\overline{x} + t(\hat{x} - \overline{x}) \in \Omega_k^+\}, \overline{x}^{s+1} = \hat{t}\,\hat{x} + (1 - \hat{t})\overline{x}, u^{s+1} := u^0, k(s+1) := k_{s+1}$, $d(s+1) = 1, s + 1 := s$ and go to 0. The next theorem is a consequence of theorems 1 and 2 and the Newton Method properties (see [5]).

Theorem 3 If $f_0(x)$ and $-f_i(x), i = 1, \overline{m}$ are convex, $f_i(x) \in C^2, i = \overline{0, m}$ and the conditions (2) - (4) are fulfilled. Then for a small enough $\varepsilon > 0$ and $0 < \gamma \leq \gamma_k \leq \frac{1}{2}$ there exists such s_0 that for $s \geq s_0$ ("hot" start). The next statements

1) $k(s) = k_{s_0} = k$ and the stepsize $t = 1$,

2) every NMBM step ("large" step), i.e., every updating of the vector u requires $0\,(\lg_2\lg_2\varepsilon^{-1})$ Newton steps,

3) the sequence $\{y^s = (x^s, u^s)\}_{s=1}^{\infty}\,(y^0 = (\bar{x}^{s_0}, e))$ converges to $y^* = (x^*, u^*)$ with the estimation

$$(11) \qquad\qquad \max\{\|x^s - x^*\|, \|u^s - u^*\|\} \le \gamma^{-s}$$

holds true.

Now we are going to consider briefly the implementation of the NMBM for a simultaneous solution of the next dual pair Linear Programming (LP) problems:

$$(12) \qquad x^* = \operatorname{argmin}\{(p, x)/r(x) = Ax - q \ge 0\}, r_i(x) = (Ax - q)_i, i = \overline{1, m}\,;\ \Omega = \{x : r(x) \ge 0\}$$

$$(13) \qquad\qquad u^* = \operatorname{argmin}\{(q, u)/A^T u = p,\ u \ge 0\}$$

where A is $m \times n$ matrix, $p \in R^n$, $q \in R^m$, $m > n$, rank $A = n, \Omega_k = \{x : r_i(x) \ge -k^{-1}, i = \overline{1, m}\}$. The MBF and $v(y, k)$, which corresponds to the problem (12) are

$$F(x, u, k) = \begin{cases} (p, x) - k^{-1}\sum_{i=1}^{M} u_i \ln(k\, r_i(x) + 1) & x \in \operatorname{int} \Omega_k \\ \infty & x \bar{\in} \operatorname{int} \Omega_k \end{cases}$$

and $v(y, k) = v(x, u, k) = \max\{-\min_{1 \le i \le m} r_i(x),\ \|p - u\Delta^{-1}(x, k)A\|,\ \sum_{i=1}^{m} u_i\,|f_i(x)|\}$,

where $\Delta(x, k) = [\operatorname{diag}(k\, r_i(x) + 1)]_{i=1}^{m}$.

The system (10) turns into the normal system of equations

$$(14) \qquad\qquad A^T k U \Delta^{-2}(x, k)A\zeta = -(p - u\,\Delta^{-1}(x, k)A)$$

where $U = [\operatorname{diag} u_i]_{i=1}^{m}$. Taking $\varepsilon = 2^{-L}(L$ - number of bits of the input data), we obtain due to theorem 3, that for any nondegenerate dual pair of the LP problems beginning at the "hot" start, one can improve (see (11)) the current approximation of at least twice ($\gamma \le \frac{1}{2}$) in $0\,(\lg_2 L)$ Newton Method steps in the worth case. Suppose that $a_{ij}, p_j, q_i, i = \overline{1, m}, j = 1, n$ are integers and

$$\max_{\substack{1 \le i \le m \\ 1 \le j \le n}}\left\{|a_{ij}|,|p_j|,|q_i|\right\} \le 2^l$$

Under this natural assumption: $l << n < m$ the input length L can be estimated by $l(m+1)^2$. So beginning at the "hot" start, which depends on the "measure" of the nondegeneracy of the primal-dual solution, one can improve the current approximation of at least twice in $0\,(\lg_2 m)$ Newton Method steps in the worth case. Therefore it seems promising to combine the universal self-concordant properties (see [3]), of the CBF which guarantees the polynomial complexity bound of the IPM, beginning at the "warm" start, with excellent MBF properties (see theorems 1, 2), which guarantees the essential improvement of this bound beginning at the "hot" start. In other words, following along the CBF trajectory from "warm" to the "hot" start, one can guarantee the improvement of the current approximation at least twice in $0(\sqrt{m})$ Newton Method steps while following along the MBF trajectory, beginning at the "hot" start, it is possible to guarantee the same improvement in $0\,(\lg_2 m)$ Newton steps in the worth case. Moreover, the system (4) is much more stable than the correspondent system which one has to solve at every step in the interior paint methods.

References

1. Dennis, J., Schnabel, R., Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice-Hall Inc., New Jersey, 1983.

2. Frisch, K.R., THe Logarithmic Potential Method of Convex Programming. Memorandum of May 13, 1955, University Institute of Economics, Oslo.

3. Nesterov, Yu, Nemirovsky, A., Self-concordant Functions and Polynomial-Time Methods in Convex Programming, CEMI Acad. of Scien., USSR, Moscow, 1989.

4. Polyak, R., Modified Barrier Functions, IBM Research Report RC 14602 No. 65413, IBM T.J. Watson Research Center, Yorktown Heights, New York, 1989.

5. Smale, S., "Newton's Method Estimates from Data at One Point", The Merging of Disciplines in Pure, Applied and Computational Mathematics (Springer-Verlag, New York-Berlin, 1986) 185-196.

6. Todd, M., Recent Developments and New Directions in Linear Programming, Technical Report No. 827, School of Operations Research and Industrial Engineering, Cornell University, New York, 1988.

ACCUMULATED APPROXIMATION
– a new method for structural optimization by iterative improvement

by

JOHN RASMUSSEN

Institute of Mechanical Engineering, Aalborg University
Pontoppidanstraede 101, DK–9220 Aalborg East, DENMARK

Summary: A new method for the solution of non–linear mathematical programming problems in the field of structural optimization is presented. It is an iterative scheme which, for each iteration, refines the approximation of objective and constraint functions by accumulating the function values of previously visited design points. The method has proven to be competitive for a number of well–known examples of which one is presented here. Furthermore, because of the accumulation strategy, the method produces convergence even when the sensitivity analysis is inaccurate.

1. Introduction

Real–life structural optimization problems are traditionally solved by sequential programming methods based on a sequence of explicit subproblems generated by sensitivity analysis. The functions involved in structural optimization, for instance stresses, displacements and eigenfrequencies, are usually highly non–linear in the space of design variables, and therefore inaccurately approximated by linear functions. On the other hand, the task of sensitivity analysis is usually too time consuming to allow for the use of higher order approximations.

A certain class of non–linear optimization problems, typically encountered in the field of structural shape optimization, is characterized by a relatively low number of design variables, eg. 10, and a medium number of constraints, eg. 100. This means that the problem from a mathematical programming point of view is a modest one. However, the objective function and constraints are implicitly defined as the result of costly numerical calculations, often performed by a finite element module. Numerical sensitivity analysis is used for the generation of an explicit subproblem, the solution of which leads to an improved design. The OASIS system by Esping [1] and the CAOS system by Rasmussen [2] are examples of implementations of this strategy. The time consumption for a typical shape optimization problem consists of more than 99% analysis and sensitivity analysis and less than 1% actual optimization.

Problems of this character are often solved using sophisticated move–limit strategies in combination with sequential linear programming (SLP) or by generating alternative first order approximations which are more conservative and for physical reasons in most cases represent the actual behavior of the structure better than linear functions do. Convex Linearization (CONLIN) by Fleury [3] and The Method of Moving Asymptotes by Svanberg [4] are examples of such algorithms. SLP, CONLIN and MMA have the common property that they base each optimization step on data evaluated in the current design point and disregard the information obtained in previous iterations.

The method of accumulated approximations (ACAP) is an improvement technique for any of the methods mentioned above; in this paper, it is demonstrated in connection with SLP. ACAP is based on the idea that, in the case of very costly function evaluations, time is well-spent on the optimizer if it reduces the total analysis time via an increased rate of convergence. Consequently, all available information should be retained in the algorithm and used if possible. The ACAP algorithm is based on a usual function approximation, linear or any other, in the current design point, but function values of previous design points are also included in the approximation in order to model the actual functions as precisely as possible with the available information. The accuracy of the approximation improves steadily as the information from previous design points is accumulated. Thus, ACAP is a very robust technique even though it can be based solely on first order information.

2. Traditional methods

In the scope of structural optimization, state functions, for instance displacement, strain and stress fields, depend on the so-called design variables, $x \in \Re^n$ which typically describe geometry, material properties etc. The state functions can only be found as the result of a costly numerical calculation and their explicit form is thus unknown. It is therefore a challenging task to perform an optimization with the purpose of finding a set of design variables which minimizes some objective, $f(x)$, eg. weight or maximum stress, while fulfilling a number of constraints, $g_i(x)$:

Minimize $\qquad\qquad\qquad\qquad\qquad\qquad f(x)$

$$(1)$$

Subject to $\qquad\qquad\qquad\qquad g_i(x) \le G_i \qquad\qquad (i=1..m)$

When solving this implicit problem by sequential linear programming, f and g_i are approximated by a succession of explicit linear functions. Functions f and g_i are evaluated at the current design point, $x^{(k)}$, and a sensitivity analysis leading to derivatives of f and g_i w.r.t. each design variable is performed. Based on this information the following explicit problem is generated:

Minimize $\qquad\qquad\qquad\qquad f(x^{(k)}) + \nabla f \cdot (x-x^{(k)})$

$$(2)$$

Subject to $\qquad\qquad\qquad g_i(x^{(k)}) + \nabla g_i \cdot (x-x^{(k)}) \le G_i \qquad\qquad (i=1..m)$

Eq. (2) is a linear problem and can be solved by means of the SIMPLEX method. It is important to notice that (2) is a mere approximation of the original nonlinear problem (1). Thus, (2) is only reliable in a small region surrounding $x^{(k)}$, the so-called trust region to which the solution is restricted. New approximations are successively generated and solved within new trust regions until convergence is achieved. Usually, the functions f and g_i are non-linear which means that a comparatively high number of iterations are required to reach the optimum. Since each iteration can be very costly, the full solution of the optimization problem is frequently a formidable task.

Sequential programming based on first order approximations is not the only way of solving structural optimization problems. Various other possibilities are available:

- Sequential quadratic programming (SQP) is based on the assumption that a quadratic approximation of the objective function will increase the convergence. In many cases of structural optimization, the objective function is in fact linear (for instance weight), in which case quadratic approximation is unlikely to help very much. Furthermore, SQP offers no solution to the problem of inaccurate sensitivities.

- Various types of line-search algorithms have been known to be very reliable. They rely on gradient evaluations to pick a search direction, but the line search itself can be based exclusively on function evaluations, ie., no sensitivity analysis is required. However, function evaluations are, as previously explained, usually rather costly.

3. Accumulated approximation

The observations of the preceding section have lead to the idea of improving the linear approximation of the objective function and constraints, ie. the n-dimensional surfaces created by the first order sensitivity analysis, by including the function values of previous design points. The modification should have no influence when the process converges steadily, but in regions of instable convergence, design points will accumulate and lead to improved accuracy of the approximation.

Let $F(x)$ be the function to be approximated (F could be either the objective function or a constraint). The accumulated approximation $P(x)$ of F is based on a usual first order approximation, $L^{(k)}(x)$, generated by sensitivity analysis and originated at the latest design point, $x^{(k)}$. In the forthcoming examples, a linear approximation is used, ie.,

$$L^{(k)}(x) = F(x^{(k)}) + \nabla F(x^{(k)}) \cdot (x-x^{(k)}) \qquad\qquad (3)$$

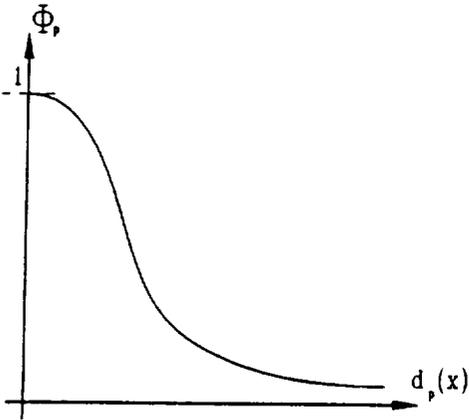Furthermore, consider the functions

Fig. 2. Influence function Φ_p.

$$d_p(x) = \sum_{j=1}^{n} (x_j - x_j^{(p)})^2 \qquad (4)$$

where the design variables x_j ($j=1..n$) are assumed to be of the same dimension. The function $d_p(x)$ is thus the squared distance from $x^{(p)}$ to any point x in n dimensions. We shall use d_p in the definition of the influence function of the p'th design point:

$$\Phi_p = e^{-\dfrac{d_p(x)}{s_p}} \qquad (5)$$

where $s_p > 0$. Φ_p has its maximum $= 1$ at $x = x^{(p)}$ and vanishes for $d_p(x) \to \infty$. The exponent is rendered dimensionless by the parameter s_p which could be considered as the square of a characteristic distance. The size of s_p is thus a measure of the sphere of influence of Φ_p. We are now ready to define the k'th accumulated approximation $P^{(k)}$ of the function F:

$$P^{(k)}(x) = \frac{L^{(k)}(x) \prod\limits_{p=1}^{k-1} [1-\Phi_p(x)] + [1-\Phi_k(x)] \sum\limits_{p=1}^{k-1} \Phi_p(x)F(x^{(p)})}{\prod\limits_{p=1}^{k-1} [1-\Phi_p(x)] + [1-\Phi_k(x)] \sum\limits_{p=1}^{k-1} \Phi_p(x)} \qquad (6)$$

Eq. (6) is constructed with the purpose of letting known functional values influence the approximation $P^{(k)}$. In regions where such values are not available, $P^{(k)}$ takes the value of the first order approximation, $L^{(k)}$. The product of terms $[1-\Phi_p(x)]$ tends to 0 near previous design points and makes sure that $L^{(k)}$ looses its influence. Similarly, the factor $[1-\Phi_k(x)]$ causes previous design points to loose their influence in the vicinity of the current point. This is the well known technique of Lagrangian interpolation. The denominator is the sum of influence functions and has its primary effect in rendering the sum of influences to unity in regions where P is influenced by several design points. Please notice that, for k=1, the convention

$$\prod_{p=1}^{0} [1-\Phi_p(x)] = 1 \qquad (7)$$

renders the first approximation $P^{(1)}(x) = L^{(1)}(x)$. The gradients of F in previous points are disposed and not used in (6). This means that, as more previous points become available, the influence of possibly inaccurate sensitivities decreases.

In choosing the characteristic distance s_p we notice that using a large s_p will cause the p'th design point to influence a relatively large region and reduce the accuracy of $P(x^{(p)})$ (p=1..k-1) because influences from other functional values than $F(x^{(p)})$ may become significant. On the other hand, relatively small values of s_p tend to generate local extrema of P at each $x^{(p)}$ (p=1..k-1) if these are far apart compared to s_p. Thus, the following recalculation of all s_p's (p=1..k-1) for each new iteration k is suggested:

$$s_p = \alpha \sum_{j=1}^{n} (x_j^{(k)} - x_j^{(p)})^2 = \alpha \, d_p(x^{(k)}) \qquad (8)$$

255

where α is a dimensionless factor. This scheme has the effect of giving points far away from the current point, $x^{(k)}$, a large sphere of influence. Points close to $x^{(k)}$ have very local influence and do not impede the local accuracy of $P^{(k)}$. For the current point $x^{(k)}$, the parameter s_k is given by

$$s_k = \alpha \, d_{k-1}(x^{(k)}) \qquad (9)$$

It is the intention of the present work to replace the usual first order approximations of the objective function and constraints by the accumulated approximation of eq. (6), thereby taking the maximum advantage of the available information.

4. Subproblem solution

Accumulated approximation of the objective function and the constraints leads to a non-linear explicit subproblem. Several methods of solving this problem have been tested, and a simple sequential linear programming approach has proved to be the more reliable. However, the implementation of a subproblem solver is slightly tricky. The initial accumulated approximations of the problem are very inaccurate due to the lack of previous design points. This means that move-limits must be imposed in order to prevent the algorithm from substantial oscillations. Furthermore, in order to obtain complete convergence for a generally underconstrained non-linear subproblem using SLP, an intelligent move-limit strategy must be employed. It is important to observe that the relative complexity of the subproblem is a minor difficulty because all functions are known explicitly and therefore inexpensive to evaluate in the numerical solution procedure. For further information on the subproblem solution, please refer to [5].

5. Numerical Example

In this section, the solution of a constrained non-linear proble: resented. As indicated in the previous sections, the ACAP algorithm has two tuning parameters, the influence parame. and a move limit, designated by T, which must be assigned values prior to the call of ACAP. This feature is similar to most other optimization algorithms. However, one of the philosophies of the accumulated approximation and the dimensionless formulation of the algorithm is to make it less sensitive to the value of such parameters, ie. we expect to be able to find a set of parameters which creates good convergence for a very wide range of problems. The ten bar truss example to be presented here has been executed with the parameters $\alpha = 0.01$ and $T = 0.15$. For α, experiments have revealed that $10^{-20} \le \alpha \le 0.10$ is a reliable interval. The convergence criterion is a constraint violation less than 0.1% and stationarity of the objective function within 0.1% in two consecutive iterations. No check of necessary optimality conditions is performed.
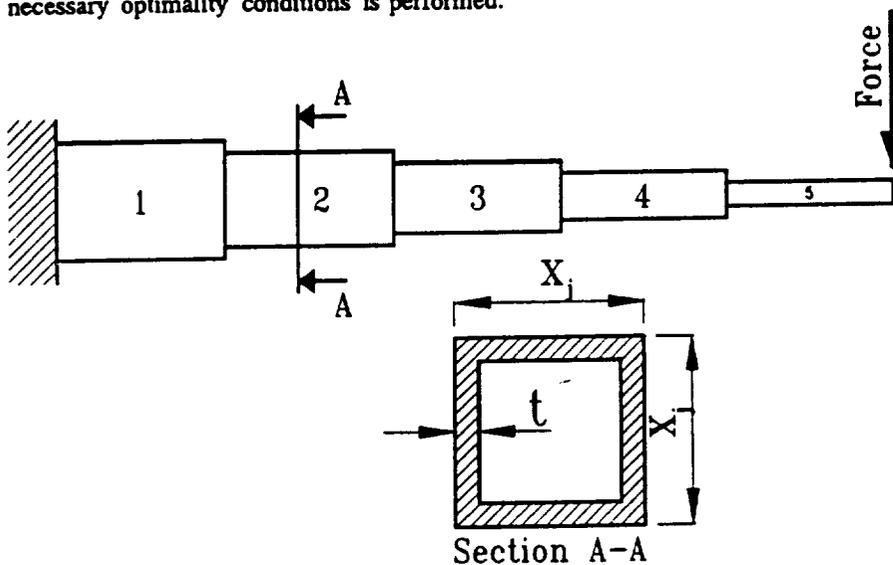
We shall consider a cantilever beam built up by 5 beam elements as shown in fig. 2. This example was originally presented by Svanberg [4] who used it as a test of the MMA-optimizer. Svanberg shows that the CONLIN optimizer by Fleury [3] does not converge at all for this problem. The present definition of the problem follows that of Svanberg in every detail.

The problem is a challenging one for a first order method because it contains more variables than active constraints at the optimum; usual SLP is unlikely to converge in this case.

The design variables are the heights of the different beam elements while the thickness t remains fixed. We shall minimize the weight with a constraint on the deflection of the beam. For a specific thickness, force and material characteristics, the

Fig. 2. Cantilever beam with 5 sizing design variables.

256

problem can be analytically formulated according to Svanberg:

Minimize
x

$$Weight = 0.0624(x_1 + x_2 + x_3 + x_4 + x_5)$$

Subject to

$$61/x_1^3 + 37/x_2^3 + 19/x_3^3 + 7/x_4^3 + 1/x_5^3 \le 1$$

Analytical solution of the problem reveals that the optimal value of the objective function is 1.34. We shall start the optimization at the feasible point $x_j^{(1)} = 5$ (j=1..5) which satisfies the constraint to equality. Tab. 1 shows the convergence of the problem.

It.no.	Weight	Violation%
1	1.560	0.0
2	1.425	13.2
3	1.359	10.0
4	1.354	3.8
5	1.331	4.8
6	1.339	0.9
7	1.337	0.9
8	1.338	0.7
9	1.339	0.3
10	1.340	0.0

Tab. 1.

Attempt	Number of iterations
1	11
2	10
3	11
4	11
5	12
6	9
7	12
8	12
9	9
10	10

Tab. 2. The number of iterations necessary to obtain convergence in ten attempts on the cantilever beam problem with sensitivities randomized by ± 20 %.

Nine iterations traversing the infeasible domain are required to reach the optimum. In this respect, the result does not compare well with the MMA method which in the best case produces convergence in just four iterations for this example. Careful tuning of the ACAP algorithm for this particular problem will reduce the necessary number of iterations to five, but that would not give a fair impression of the performance of the algorithm in the general case. In any case, ACAP is a significant improvement of SLP which does not converge at all for this problem. The competitiveness of the algorithm is further emphasized by a number of other test examples. For a further description of these, please refer to [5].

6. Influence of sensitivity accuracy

Barthelemy & Haftka [6] and Pedersen et. al. [7] have shown that, for certain types of structures, sensitivities based on finite difference may suffer from serious lack of accuracy. The ACAP algorithm refines the approximation of the objective function and constraints as the number of previous points increase. This means that the influence of the sensitivity analysis on the optimization decreases steadily and thus reduces the error caused by lack of accuracy of the sensitivity analysis.

The influence of the sensitivity accuracy has been studied by means of the cantilever beam example. The sensitivities of the objective function and constraints are artificially disturbed by individually multiplying them by a random number in the range [0.8 – 1.2], ie., a disturbance of ± 20%, before passing them on to the ACAP optimizer. Tab. 2 shows the necessary number of iterations in ten attempts. The cantilever beam problem was solved in 9 iterations with accurate sensitivities. Experiments indicate that the convergence remains stable up to a disturbance of ± 40%. Beyond this limit, the process may or may not converge.

10. Conclusion

The ACAP approximation generates a significant improvement of SLP and produces convergence in cases where neither SLP nor CONLIN converge.

For the example presented here, the MMA-method has shown faster convergence toward a feasible solution. This is most likely due to the fact that MMA as a generalization of convex linearization is a far better approximation of the true structural problems than SLP is. Thus, MMA starts out with a good approximation whereas, in the case of ACAP, such an approximation has to be generated along the way.

The ACAP approach is not limited to sequential linear approximation. Other approximation types, MMA for instance, may be subjected to improvement by the same simple strategy.

ACAP reduces the influence of possible inaccurate sensitivities. Stable convergence is observed for the cantilever beam problem with random errors of up to ± 40% introduced in the sensitivity analysis. However, inaccurate sensitivities may increase the necessary number of iterations in order to reach the optimum.

The ACAP approximation (eq. 6) is a "first attempt" to include previous design points into the problem. Interpolation techniques is a vivid area of research and better techniques may be available. The ACAP interpolation has several drawbacks of which the worst is that it is practically impossible to differentiate it analytically.

The algorithm is devised to allow the basic linear approximation $L^{(k)}(x)$ (eq. 3) to be replaced by any other first order approximation type, eg. convex approximation. The possibly positive influence of such a replacement is yet to be investigated. The flexible construction of the algorithm and the direct solution strategy furthermore enables the inclusion of equality constraints because these are handled naturally by the SIMPLEX subproblem solver.

References

1. B. Esping:
   The OASIS Structural Optimization System
   Computers and Structures, Vol. 23, No. 3, pp. 365–377 (1986).

2. J. Rasmussen:
   The Structural Optimization System CAOS
   To Appear, Structural Optimization, Vol. 2 (1990).

3. C. Fleury:
   Structural Optimization: a New Dual Method Using Mixed Variables
   Int. J. Num. Meth. Eng., Vol. 23, pp. 409–428 (1986).

4. K. Svanberg:
   The Method of Moving Asymptotes – a new method for structural optimization
   Int. J. Num. Meth. Eng., Vol. 24, pp. 359–373 (1987).

5. J. Rasmussen:
   Structural Optimization by Accumulated Function Approximation
   Report No. 20, Institute of Mechanical Engineering, Aalborg University, June 1990. Submitted for publication in the International Journal for Numerical Methods in Engineering.

6. B. Barthelemy & R.T. Haftka:
   Accuracy Analysis of the Semi–analytical Method for Shape Sensitivity Calculation
   Paper presented at the AIAA/ASME/ASCE 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, Virginia, April 1988.

7. P. Pedersen, G. Cheng & J. Rasmussen:
   On Accuracy Problems for Semi–analytical Sensitivity Analyses
   Mechanics of Structures and Machines, Vol. 17, No. 3, pp. 373–384 (1989).

OPTIMAL GLASS-CERAMIC STRUCTURES
Components of Giant Mirror Telescopes

Hans A. Eschenauer
Research Laboratory for Applied Structural Optimization
at the
Institute of Mechanics and Control Engineering
University of Siegen
Siegen, FR Germany

Summary:

Detailed investigations are carried out on optimal glass-ceramic mirror structures of terrestrial space technology (optical telescopes). In order to find an optimum design, a nonlinear multicriteria optimization problem is formulated. "Minimum deformation" at "minimum weight" are selected as contradictory objectives, and a set of further constraints (quilting effect, optical faults etc.) is defined and included. A special result of the investigations is described.

1. Introduction

Optical astronomy includes observations in the spectrum from near ultra violet via the visible light to infrared. This wave-length range will remain important in future as the physical nature of extremely distant space projects can often be observed only within the range of the optical spectrum. At present, the efficiency of terrestrial astronomy is determined by the 3 to 5-m-telescopes. In recent years light detectors used for evaluating the results have been improved so much (up to an efficiency of 100%) that these instruments have reached their efficiency limits. Thus, further discoveries can only be made by collecting more light. For this, large telescopes with effective mirror-apertures of more than 8 m Ø are necessary. That means, the traditions in telescope design have to be replaced by new optical and technical concepts [1]. All over, the world various types of telescopes are in planning, under development or already under construction (the latter case applies to the 10-m-Keck-Telescope of the University of California, Berkeley). Though, an important problem is that the costs for constructing and erecting such giant telescopes will increase astronomically. The following two fundamental demands have to be met:

- observance of an extraordinarily high precision of the mirror surface (<50 nm).

- minimization of the costs and thus giving the guarantee that such a project can be financially supported.

When dealing with an optimization problem, one should principally proceed according to the "Three-Columns-Concept". The first step is the theoretical formulation of the optimization problem regarding all relevant requirements on the structure. Then, the sub-problems "structural modelling" and "optimization modelling" have to be solved. After selecting an appropriate optimization algorithm, this algorithm is linked with the structural model and the optimization model so that a closed optimization loop is formed [2].

2. Structural Modelling and Structural Analysis

This paper deals with some problems of minimal mirror designs as elements of an entire mirror. The mirror elements are made of glass-ceramic materials (quartz glass. zerodur) loaded by pressure and temperature.

Fig. 1 illustrates the structure of a circular mirror plate with the following design parameters:

- mirror shape (circle, rectangle. hexagon).
- core cell structure (quadratic. triangular. hexagonal),
- cell size or rib distance.
- height of a cell structure.
- thickness of the layers.
- thickness of the boundary stiffening.
- arrangement of the supports.
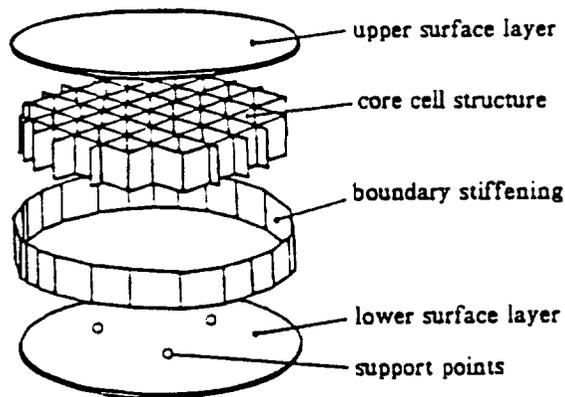
Fig. 1: Design parameters of a mirror plate

The following requirements on the material have to be fulfilled for mirror structures:

- thermal expansion coefficient as small as possible.
- homogeneity of the themal expansion coefficient over the material volume.
- pore-free surface.
- corrosion-resistance.
- no hygroscopicity.
- good mechanical workability for exact surfaces.
- chemical durability in the case of separation and removal of the reflecting layer.
- thermal stability of the material structure up to the transformation temperature (transition from a solid condition into the viscous fusibility range).

With the above-mentioned requirements we take a porous. but isotropic. linear-elastic material as a basis for the structural analysis calculations. In addition to the FE-method. two analytical methods have been applied [3].

a) Thin circular plate with variable thickness point-supported in the middle
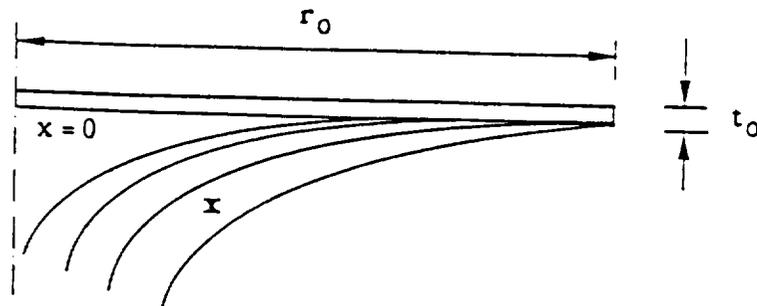
Fig. 2: Circular plate with variable thickness

The following approach for the wall thickness curve is chosen (Fig. 2):

$$t(r) = t_0 \left[\frac{r}{r_0} \right]^{-x/3} \tag{1}$$

with x = thickness parameter.

       r_0 = outer radius of the plate.

       t_0 = plate thickness at the outer radius.

Thus, the differential equations of the plate read as follows:

$$\Delta\Delta w + \frac{K'(r)}{K(r)} \left[2w''' + (2+\nu)\frac{1}{r} w'' - \frac{1}{r^2} w' \right] + \frac{K''(r)}{K(r)} \left[w'' + \frac{\nu}{r} w' \right] = \frac{p}{K(r)} \tag{2}$$

where $' \hat{=} \frac{\partial}{\partial r}$, and the variable plate stiffness $K(r) = K_0 \left[\frac{r}{r_0} \right]^{-x}$, $K_0 = \frac{E\, t_0^3}{12(1-\nu^2)}$.

b) Thin point-supported circular plate on a concentric circle

Fig. 3 shows the arrangement of point-supports for a circular plate with constant plate thickness
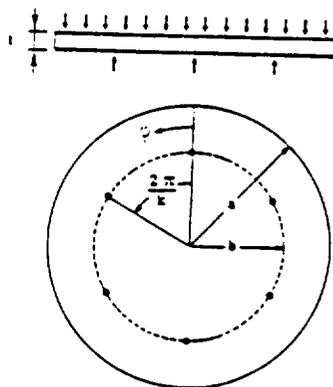
Fig. 3: Point-supported circular plate on a circular ring

The differential equation of a point-supported plate is given by

$$\Delta\Delta w = \frac{F}{K\pi a^2} - \frac{F}{Kk} \sum_{j=1}^{k} \frac{\delta(r-b)}{b} \; \delta\left(\varphi - \frac{2j\pi}{k} \right) \tag{3}$$

with δ = DELTA-functions.

261

Development by Fourier-series

$$\sum_{j=1}^{k} \delta\left(\varphi - \frac{2j\pi}{k}\right) = \frac{a_0}{2} + \sum_{m=1}^{\infty} a_m \cos k m \varphi$$

with

$$a_m = \frac{1}{\pi} \sum_{j=1}^{k} \int_0^{2\pi} \delta\left(\varphi - \frac{2j\pi}{k}\right) \cos k m \varphi \, d\varphi = \frac{k}{\pi} \, ,$$

leads to

$$\Delta\Delta w = \frac{F}{K\pi a^2} - \frac{F}{K 2\pi b} \delta(r-b) - \frac{F}{K\pi} \frac{\delta(r-b)}{b} \sum_{m=1}^{\infty} \cos k m \, \varphi. \qquad (4)$$

This solution yields with the separation approach

$$w(r, \varphi) = \sum_{m=1}^{\infty} w_m (r) \cos k m \, \varphi : \qquad (5)$$

$$m = 0: \qquad \frac{1}{r}\frac{d}{dr}\left\{r\frac{d}{dr}\left[\frac{1}{r}\frac{d}{dr}\left(r\frac{dw_0}{dr}\right)\right]\right\} = \frac{F}{K\pi a^2} - \frac{F}{K 2\pi b} \, \delta(r-b) \, , \qquad (6a)$$

$$m \geq 1: \qquad \left(\frac{d^2}{dr^2} + \frac{1}{r}\frac{d}{dr} - \frac{k^2 m^2}{r^2}\right)^2 w_m = \frac{F}{K 2\pi b} \, \delta(r-b) \qquad r = b, \qquad (6b)$$

$$\left(\frac{d^2}{dr^2} + \frac{1}{r}\frac{d}{dr} - \frac{k^2 m^2}{r^2}\right)^2 w_m = 0 \qquad r \neq b. \qquad (6c)$$

For a plate with r = 150 mm. the deflections and corresponding analysis methods are compared in Fig. 4.
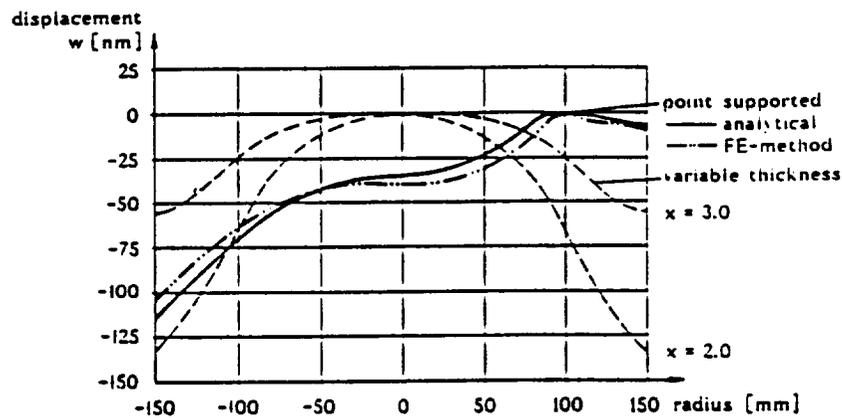
Fig. 4: Comparison of the displacements of a circular plate

3. Optimization Modelling and Procedure

The present structural optimization task is considered as a Multicriteria-Optimization-Problem (MC-Problem). In this case. a design variable vector x is to be found which makes the m components of the objective function vector f as small as possible while fulfilling all constraints. A MC-Problem can mathematically be defined by the following model formulations [2]:

262

a) **Model 1**: Continuous. deterministic MC-Problem

$$\underset{x \in \mathbb{R}^n}{\text{"Min"}} \ \{ f(x) \ : \ h(x) = 0. \ g(x) \leq 0 \}$$

with the following symbols

$\mathbb{R}^n$	set of real numbers.
f	vector of m objective functions (weight. surface quality),
$x \in \mathbb{R}^n$	vector of n design variables
g	vector of p inequality constraints (quilting effect. failure criteria. bounds),
h	vector of q equality constraints (e.g. system equations for determining stresses and deformations),

and

$$X := \{ x \in \mathbb{R}^n : h(x) = 0. \ g(x) \leq 0 \}$$

"feasible" domain where $\leq$ is to be interpreted for each single component.

b) **Model 2**: Discrete. deterministic MC-Problem:

$$\underset{x \in X_d}{\text{"Min"}} \ \{ f(x) \}$$

with the discrete design space

$$X := \{ x \in \mathbb{R}^n \mid x_j \in X_j \ : \ j = 1. \ \dots \ N \ ; \ g(x) \leq 0. \ h(x) = 0 \}$$

and the N sets of discrete values

$$X_j = \{ x_j , x_j , x_j , \dots x \ \}. \ X_j \subset \mathbb{R} \ \forall \ j = 1. \ \dots \ N$$

with n_j number of discrete values of the j-th design variable.

c) **Model 3**: Stochastic MC-Problem

$$\underset{y \in Y}{\text{"Min"}} \ \left\{ f(y) \mid P[f(y)] = r^f. \ P[g(y) \leq 0] = r^g \right\}$$

with

y	vector of N random variables (loads. dimensions. characteristic values of the material) including design variables.
Y	vector of the expected values of the N random variables.
$P[\ ]$	probability.
$r^f. \ r^g$	vectors of the m or q reliabilities concerning the objective and inequality constraints.

For the optimal layout it is especially important to apply the appropriate optimization model. The optimization problem itself is defined by two main criteria (objective functions). namely the "weight of the mirror structure" and the "surface accuracy". The quilting-effect as a special manufacturing error is considered as an inequality constraint. Here. the core cell of the mirror structure is slightly deformed by the

polishing load. This pillow-shaped deformation leads to a periodical deformation of the total mirror surface [1]. Due to the probabilistic material values (Weibull-distribution of strength) and the numerous discrete design parameters (number of cells and layers). the optimization models 2 and 3. respectively, have been established and special optimization algorithms are applied in order to find the most appropriate solutions.

4. Optimal Designs of Mirror Plates

Optimization calculations were carried out for an orthotropic plate with a quadratic cell structure under combined pressure and temperature loads. Some results of the optimization process are shown in Figure 5 . whereas the Pareto-efficient curves for a zerodur and a quartz glass mirror plate are compared. The curves of the zerodur mirror plate and the quartz-glass mirror plate with the same weight and the same loads show that the zerodur mirror plates provide a four times higher surface accuracy [1].
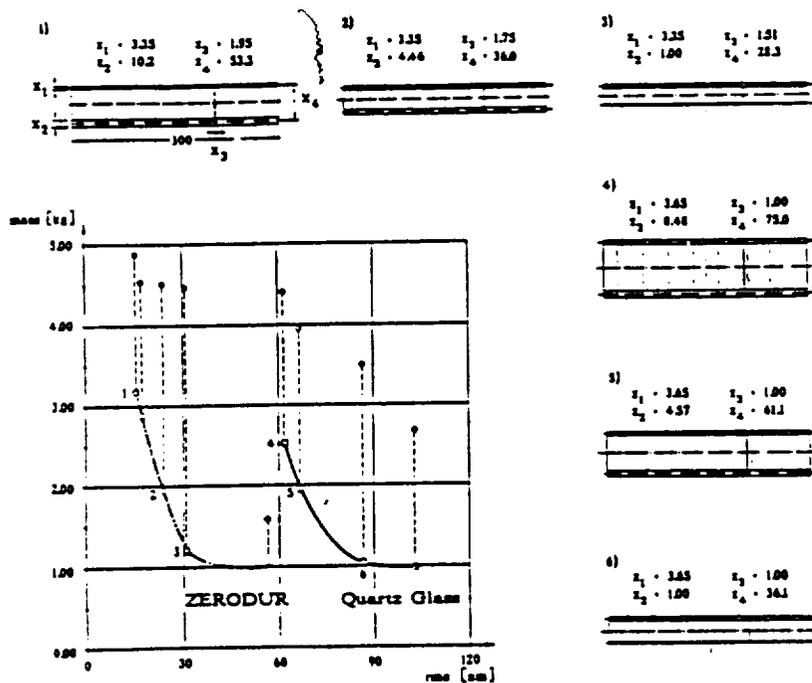
Fig. 5: p-efficient solutions of a mirror plate

Further investigations have been carried out with regard to the optimal ratio of support radii and the optimal number of support points as a function of the rms-value of surface accuracy. The calculations resulted in an optimal ratio of support circle of $r_s / r = 2/3$ and number of support points $n_{opt} = 3$ for the chosen plate. The first mentioned result approximately agrees with the one of [3]. that means that the additional temperature load in our calculations is of no great relevance for these values.
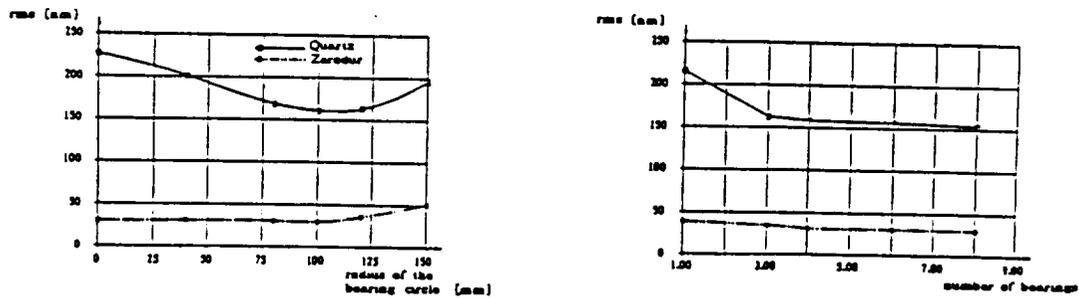
Fig. 6: Optimal ratio of support radii and optimal number of support points

5. Conclusion

The task of structural optimization is to support the engineer in searching for the best possible design alternatives of specific structures. The "best possible" or "optimal" structure is that which highly corresponds to the designer's desired concept and his objectives while at the same time meeting the functional, manufacturing and application demands. By means of a special optimization procedure it is possible to establish more realistic and more reliable models to improve the computation of optimal designs of a number of technically relevant structures. Results of computation are shown for a special mirror component made of glass-ceramics as a part of a large mirror.

References

[1] Eschenauer, H.A.: Telescope Designs - Structural Optimization of Mirror Components. Proceedings of the 1. Convention of the IES "Engineering Achievments and Future Challenges". Singapore. 17-19 may 1990. pp. 145-154.

[2] Eschenauer, H.A.. Koski. J.. Osyczka. A.: Multicriteria Design Optimization. Berlin. Heidelberg, New York, London. Paris. Tokyo. Hongkong: Springer-Verlag, 1990

[3] Budianski. M.P.. Nelson. I.E.: Analysis of mirror supports for the University of California Ten Meter Telescope. Proceedings of the International Society for Optical Engineering. Berkely, Ca.. 1983

Employment of C.B. models for non-linear dynamic analysis

M.R.M. Klein * Ph. Deloo *** A.Fournier-Sicre **

Abstract:

The non-linear dynamic analysis of large structures is always very time, effort and CPU consuming. Whenever possible the reduction of the size of the mathematical model involved is of main importance to speed up the computational procedures. Such reduction can be performed for the part of the structure which perform linearly. Most of the time, the classical Guyan reduction process is used. For non-linear dynamic process where the non-linearity is present at interfaces between different structures, Craig-Bampton models can provide a very rich information, and allow easy selection of the relevant modes with respect to the phenomenon driving the non-linearity.

The paper presents the employment of Craig-Bampton models combined with Newmark direct integration for solving non-linear friction problems appearing at the interface between the Hubble Space Telescope and its Solar Arrays during in-orbit manoeuvres. Theory, implementation in the FEM code ASKA and practical results are shown.

1. Craig-Bampton models background

The general equation of motion of a structure can be written as

$$M\ddot{q} + C\dot{q} + Kq = F(t) \tag{1}$$

Considering the interface of the structure with other structures, we can split the dof's in internal dof's i and interface dof's j and write :

$$\begin{bmatrix} M_{ii} & M_{ij} \\ M_{ji} & M_{jj} \end{bmatrix} \begin{pmatrix} \ddot{q}_i \\ \ddot{q}_j \end{pmatrix} + \begin{bmatrix} K_{ii} & K_{ij} \\ K_{ji} & K_{jj} \end{bmatrix} \begin{pmatrix} q_i \\ q_j \end{pmatrix} = \begin{pmatrix} 0 \\ F_j(t) \end{pmatrix} \tag{2}$$

where without loss of generality we have assumed that only interface loads are applied and neglected the damping matrix.

The Craig-Bampton approach consists of developing the displacements q on the basis of the static interface modes Φ_j and the elastic modes φ_p of the structure clamped at the interface j :

$$q = \Phi_j\, q_j + \varphi_p\, \eta_p = \Psi\, X \tag{3}$$

where

$$\Psi = (\Phi_j\ \varphi_p) \quad \text{and} \quad X = \begin{pmatrix} q_j \\ \eta_p \end{pmatrix} \tag{4}$$

$$\Phi_j = \begin{pmatrix} \Phi_{ij} \\ I_{jj} \end{pmatrix} \quad \text{where} \quad K\,\Phi_j = 0 \tag{5}$$

* European Space Agency / European Space and Technology Center ESA/ESTEC
  Postbus 299, 2200AG Noordwijk, The Netherlands.
** ESA/ESTEC, Space Telescope Solar Array Manager
*** Consultant, Ikoss Gmbh, Waldburgstrasse 21 D-7000-Stuttgart 80, Federal Republic
  of Germany

$$\varphi_p = \begin{pmatrix} \varphi_{ip} \\ 0 \end{pmatrix} \quad where \quad (K_{ii} - \omega^2 M_{ii}) \, \varphi_{ip} = 0 \qquad (6)$$

As a consequence of such transformation, the initial equations of motion get transformed

to

$$\begin{bmatrix} \overline{M}_{jj} & \overline{M}_{jp} \\ \overline{M}_{pj} & m_{pp} \end{bmatrix} \begin{pmatrix} \ddot{q}_j \\ \ddot{\eta}_p \end{pmatrix} + \begin{bmatrix} \overline{K}_{jj} & 0 \\ 0 & k_{pp} \end{bmatrix} \begin{pmatrix} q_j \\ \eta_p \end{pmatrix} = \begin{pmatrix} F_j(t) \\ 0 \end{pmatrix} \qquad (7)$$

where : $\overline{K}_{jj}$ is the stiffness matrix statically reduced to the j interface,

$\overline{M}_{jj}$ the mass matrix reduced at the interface according to the Guyan reduc-

tion   concept,

k_{pp} and m_{pp} the generalized stiffness and mass matrices,

$\overline{M}_{jp}$ $= \Phi_{ij}^T M_{ii} \, \varphi_{ip} + M_{ji} \, \varphi_{ip}$ is the modal participation factor matrix.

It must be noticed that both the physical and CB representation of the structure are mathematically equivalent, and both sets of matrices represent the free–free structure.

Modal selection can be performed based on the well known "Effective Modal Mass" concept with respect to the j interface. However, modes with low interface force contribution but describing internal spacecraft dynamics which is of interest have to be kept in addition to the modes dominating the interface response. This selection reduces significantly the size of the CB models, which is one of their advantages.

Furthermore, CB models avoid the usual stiffness and low frequency mass truncation problems, since the interface stiffness is included in the $\overline{K}_{jj}$ matrix and the total mass in the $\overline{M}_{jj}$ matrix.

In the same way FEM models are assembled, Craig–Bampton models can also be assembled, the connection always taking place using the physical dofs. The reduced size of all individual CB models reduces the size of the system CB model to a minimum set of modes which produces the most efficient representation of the spacecraft. Currently 3000 to 10000 dof FEM models can be efficiently reduced to 30 to 1000 Craig–Bampton freedoms.

The final assembled system model can if necessary be analyzed in free–free conditions. In this case, to ensure simplest correct recovery of interface forces and of stresses, most of the spacecraft free–free modes have to be used. Using Craig–Bampton models reduces this set to the most significant modes for the problem.

Another advantage of those models is their relationship with the clamped modes identified during testing, which provides a good engineering approach.

2. Handling non–linearities

Craig–Bampton models intrinsically address linear phenomenon, since constant material properties and usual FEM small deformation and displacement theory is used. However, there is room for non–linearity at the interface freedoms. This can occur as variable stiffness, connected/disconnected status of some freedoms and variable loads function of some interface freedom responses.

These limited cases are, however, representative of many practical interface effects, like gapping, non–linear tension–compression contact, torque limiter effects, etc..

These processes introduce time dependent perturbations in the equations of motion, limited to the interface freedoms for the CB mass and stiffness matrices, which shall be ac-

counted for in the time integration process. They lead to a modification of the clamped condition of the interface freedoms as considered in the clamped modes. However, there is no contradiction in releasing CB interface dof's since we have seen that the CB model actually represents the complete free-free structure.

Since non-linearities are present in the behaviour of the final assembled system, standard time integrations methods as modal superposition cannot be applied anymore. A direct integration scheme has to be selected. This can be performed without problems since the CB model is just another representation of the physical FEM model. Integrating equations in a modal space by direct integration, although being unusual, is not contradictory. It must be noticed that the reduction of number of equations to integrate, usually resulting from the modal superposition approach, has actually now been performed at the Craig-Bampton model level. Thus this approach is not penalizing in terms of size reduction.

One of the most problematic aspects of usual direct integration approach (i.e. with physical dof matrices) is the damping aspect. Indeed, although modal damping values can be infered from engineering experience or derived from testing, the implemenation of these (in form of Raleigh series matrices) is quite cumbersome and seldom performed. Using the CB approach on the other hand, these values appear naturally in the equations of motion.

Indeed, the physical damping matrix C transforms into the CB space as ...

$$\Psi^T C \Psi = \begin{bmatrix} \Phi_j^T C \Phi_j & \Phi_j^T C \varphi_p \\ \varphi_p^T C \Phi_j & \varphi_p^T C \varphi_p \end{bmatrix} \quad (8)$$

In case of static determinate interface, Φ_j are the rigid body modes which are assumed not damped, hence $C\Phi_j = 0$ and $\Psi^T C \Psi$ bolts down to

$$\Psi^T C \Psi = \begin{bmatrix} 0 & 0 \\ 0 & [2m\xi\omega] \end{bmatrix} \quad (8)$$

where $[2m\xi\omega]$ is a diagonal matrix and the ξ are the percentages of critical viscous damping ratios of the clamped interface modes. Correctness of this assumption can easily be verified by considering the global CB system equations and clamping the system at the interfaces. The modal response equations as derived directly from the physical matrices are identical to those derived from the CB model equations.

In case of non static-determinate interfaces, the 3 other terms of the damping matrix can be either ignored, or handled in the usual way, i.e. using proportional damping $C = \alpha K + \beta M$, Raleigh series or discrete dampers. The direct integration algorithm will have no problem handling damping-coupled equations.

3. Implementation in ASKA

The methodology described has been implemented in the general purpose FEM package ASKA distributed by IKOSS. Considering the application planned (torque limitation and slippage induced by a friction brake on a shaft) only static determinate interface was implemented in a first step.

The standard Newmark integration scheme has been used, together with matrix manipulation and partition/merge tools to handle the interface condition charges.

The freedoms for which non–linarities will occur are attached to 2 nodes. These freedoms are either disjoint but subject to self equilibrating loads simulating friction, or connected. In this last case, the combined equation of motion are merged into one unique dof and constrain relations are applied to the other dof, thus maintaining constant the dof pattern of the dynamic system.

Equilibrium iterations are performed at each time step, and variable time step is used (bi-section method) to ensure a proper convergence to the change of state of the non–linear freedoms (i.e. slippage/no–slippage). This change of state is defined considering the system state vector using user–written Fortran routines which can be naturally interfaced with ASKA. The considered change conditions are :

No slippage — until maximal torque is reached in shaft;

Once slippage occured — constant maximal balanced torque applied to both disconnected parts of the shaft;
— slippage until relative velocity of 2 shaft parts comes back to zero.

4. Application to HST

The procedure including all features described above has been applied in the frame of the Hubble Space Telescope project, where the European Space Agency (ESA) is responsible for the Faint Object Camera and the Solar Arrays. Whatever sky region the HST is looking at, the SA is designed to be oriented toward the sun to get the maximal power. When the optimal position is obtained, a brake is released to prevent any modification of the SA position.

The deployment of the SA in orbit occurs when the HST is held on the Space Shuttle remote manipulator system (RMS). After deployment of the SA, the HST is still on the remote manipulator system for a few hours before release. During that period the Shuttle performs manoeuvres to hold attitude, firing thrusters. This induces loads on the SA. Since the HST–SA interface is basically a 6 dof's joint, the interaction can be described by the interface accelerations. A CB model of the SA has been developed by ESA to assess the in orbit load level when subject to such manoeuvres

Figure 1 shows the complete 3000 dofs FEM model and some of the clamped interface modes. The location of the brake between HST and SA is sketch–wise explained in Figure 1a. The main modes of the SA in clamped interface conditions are identified in Table 1 in terms of modal masses (percentage of rigid body mass corresponding terms); those 21 modes were used to build the SA CB model. Figure 2 shows free–free modes computed with this model. Notice that although blanket aspects are completely accounted for, only boom motions are visualised since those are used later on to compute boom bending moments. These booms are indeed the most delicate mechanical parts of the deployed SA.

The brake slippage procedure as been tested using the angular acceleration around the SA Y axis shown in Figure 3 applied at the HST / SA interface. The torque actually seen by the SA on the other side of the brake is shown in Figure 4, and the related SA angular acceleration is in Figure 5. The inspection of the response of the system shows that the change of configuration/loading procedure is working fine. The interactive torque never exceeds 0.5 Nm which is the specified maximum brake torque in this test. The change of configuration/loading is smooth, however it is interesting to note that a shock is induced

in the system (see Figure 5) when the brake starts slipping. The SA oscillates about its axis at a frequency close to 1 Hz. The mode which is excited is equivalent to the free-free mode 13 of the SA model presented in Figure 6. It shows a large rotation of the drum while the boom tip hardly moves.

Regarding in-orbit actual manoeuvres, classical rigidly connected linear CB analysis using modal superposition method applied to the free-free modes has shown that for some types of manoeuvres the interface torque overshoot largely the torque the brake could withstand without slippage. In addition loads in the SA booms were exceeding boom bending moment allowables. Thus the analysis had to be refined. In order to account for this effect, transient analyses have been performed using the procedure described above. This allowed changes of the brake connection from rigid state to slippage state and vice-versa as many times as requested during each computer simulation. A more realistic behaviour could be simulated, slippage of the SA evaluated and subsequent reduction of loads in the booms demonstrated.

Figure 6 shows results related to in-orbit event PO4 for the +V2 wing. As can be seen, the interface torque does not overshoot the maximun brake-through torque (0.87 Nm) and slippage occurs after this limit is reached. Table 2 shows a summary of 17 worst load cases which were analysed. Substantial load reduction occurs for the solar array booms, with relative low array rotation which does not affect significantly the SA power performances. The computing time per run was quite reasonable (20 minutes on VAX 8650 for a response of 40 sec) considering the highly non-linear aspects of the brake conditions, and allowed a satisfactory computation of all cases; it must be kept in mind that these results reflect the information related to a 3000 dofs model, which itself could never be used for such complex analyses.

5. Conclusions

Craig-Bampton dynamic models enable a very rich information to be condensed in a very compact format. In combination with direct integration they are able to handle non-linearities related to interface freedoms. Moreover, they can use in a straightforward way the usual modal damping.As such, they build a very efficient tool to analyse certain class of non-linear problem with minimum computer resources.

The application to the HST deployment manoeuvre scenario was very successful; it showed substantial SA boom load reduction and allowed to estimate the SA rotation.

This promising procedure is now in development for application to more general cases involving non static-determinate interfaces.

References

1. STSA Deployed Finite Element Model E1; P.Deloo, J.Reynolds, M.Klein.
   TN-SA-0011 Vol.1 Model description, Vol.2 Modal properties and Reduced Model, Vol.3 Boom Force Recovery Methodology. ESA-ESTEC 1990
2. Procedure to perform a non-linear integration of a Craig-Bampton model in ASKA. P.Deloo, M.Klein TN-SA-0018 ESA-ESTEC Dec.1990.

Figure 1: FEM model and some related main modes.
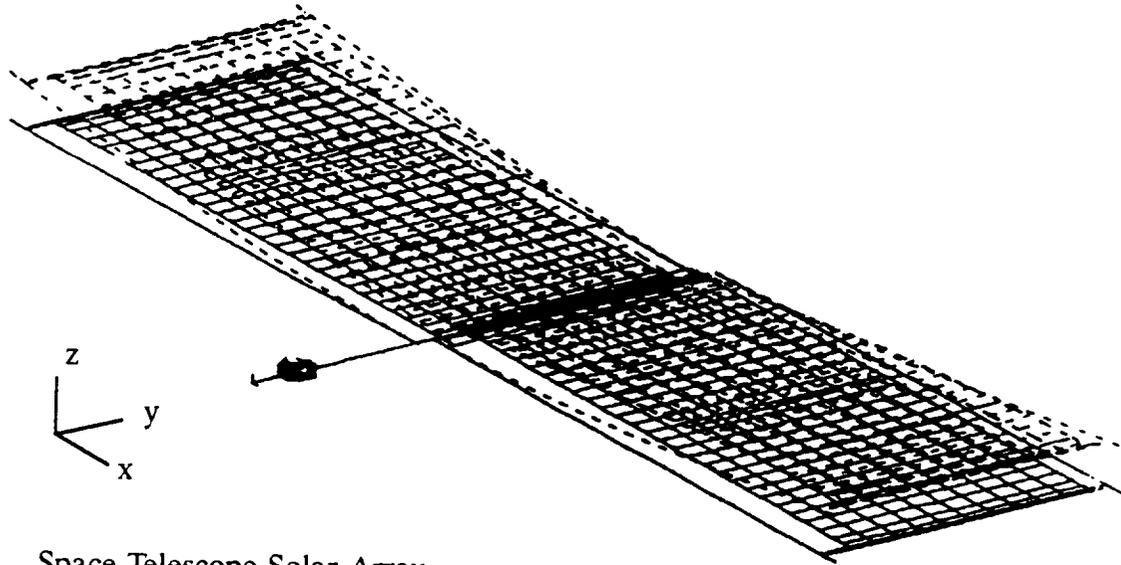
Space Telescope Solar Array
Modal deformation Mode 2 Freq. 0.086614 Hz
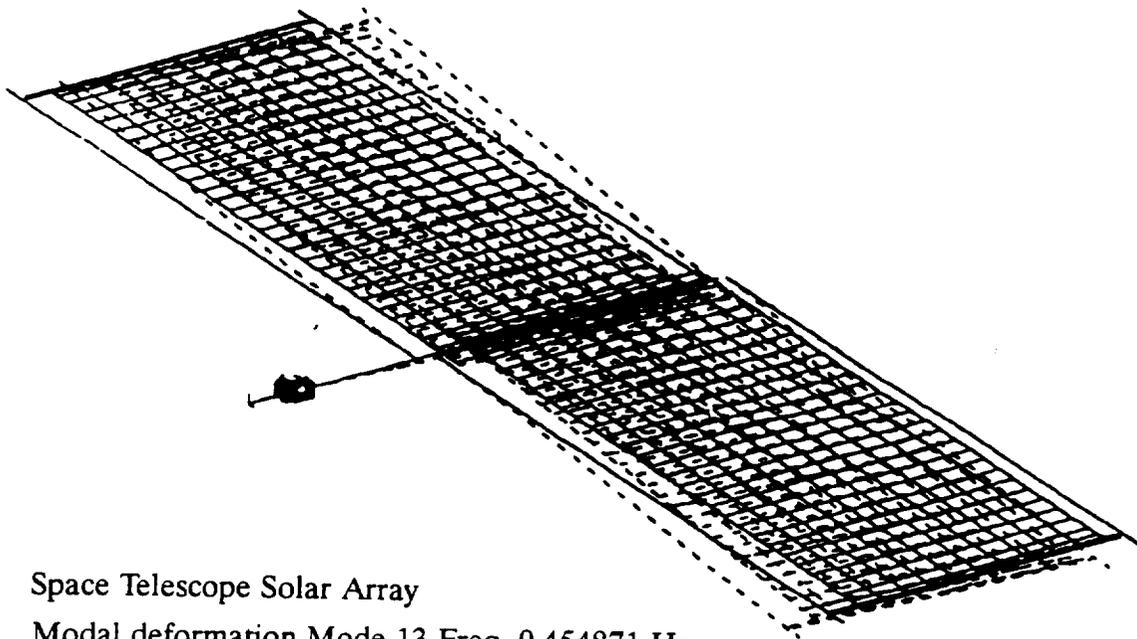
Space Telescope Solar Array
Modal deformation Mode 13 Freq. 0.454871 Hz
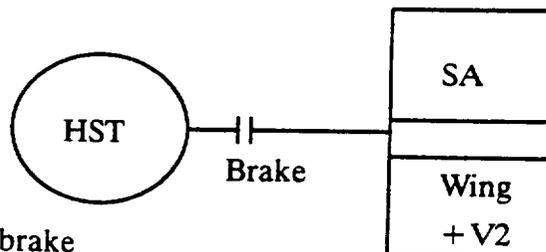
Figure 1a : location of the brake

EIG. FREQ.	NO	MX(%)	MY(%)	MZ(%)	IX(%)	IY(%)	IZ(%)
0.085713	1	0.000000	0.000000	0.001612	0.001868	99.100212	0.000001
0.086614	2	0.001069	0.000000	27.330700	31.494280	0.005601	0.000917
0.226470	3	0.000003	0.000000	0.029525	1.814183	0.000000	0.000076
0.315740	7	0.000281	0.000001	3.737133	3.941284	0.000141	0.000415
0.454371	13	25.777058	0.000170	0.000002	0.010451	0.002712	94.543091
0.541241	16	0.000193	0.000004	1.564985	2.401618	0.000002	0.000630
0.703183	27	3.463858	0.000000	0.000092	0.000048	0.000030	1.021423
0.763398	33	0.000063	0.000008	0.905543	1.022478	0.000017	0.000065
0.940869	43	0.000359	0.000039	0.775949	2.121378	0.000086	0.000125
0.983912	45	0.001190	0.000131	2.521466	5.759458	0.000284	0.000377
1.032549	49	0.000291	0.000035	0.692529	1.280281	0.000074	0.000084
1.058607	51	0.000444	0.000053	1.153723	1.805675	0.000107	0.000124
1.092661	53	0.014043	0.001497	15.655893	39.598808	0.003226	0.003325
2.684306	156	39.684685	0.000099	0.003183	0.001318	0.000882	3.694604
4.178203	251	0.000007	0.000044	3.512241	1.282615	0.000041	0.000000
4.194105	253	0.000006	0.000030	1.898331	0.696061	0.000030	0.000000
4.199246	255	0.000010	0.000048	2.741392	1.005311	0.000050	0.000000
4.227032	257	0.000008	0.000039	2.118525	0.789823	0.000041	0.000000
5.386452	320	0.000288	14.689480	0.000000	0.000000	0.000001	0.000000
5.466548	327	2.170738	0.004099	0.001671	0.000341	0.003833	0.068066
5.867115	349	0.000139	8.441351	0.000000	0.000000	0.000000	0.000022

Table 1 : Main modes of the FEM model, in terms of rigid mass percentages.

DISPLACEMENT RESULTS FILE:E1_RED28L7.DIS
LOADCASE:7 VERSION ASKPAT:2.2-6 POSTFILE:E1_RED28.POA

DISPLACEMENT RESULTS FILE:E1_RED28L8.DIS
LOADCASE:8 VERSION ASKPAT:2.2-6 POSTFILE:E1_RED28.POA
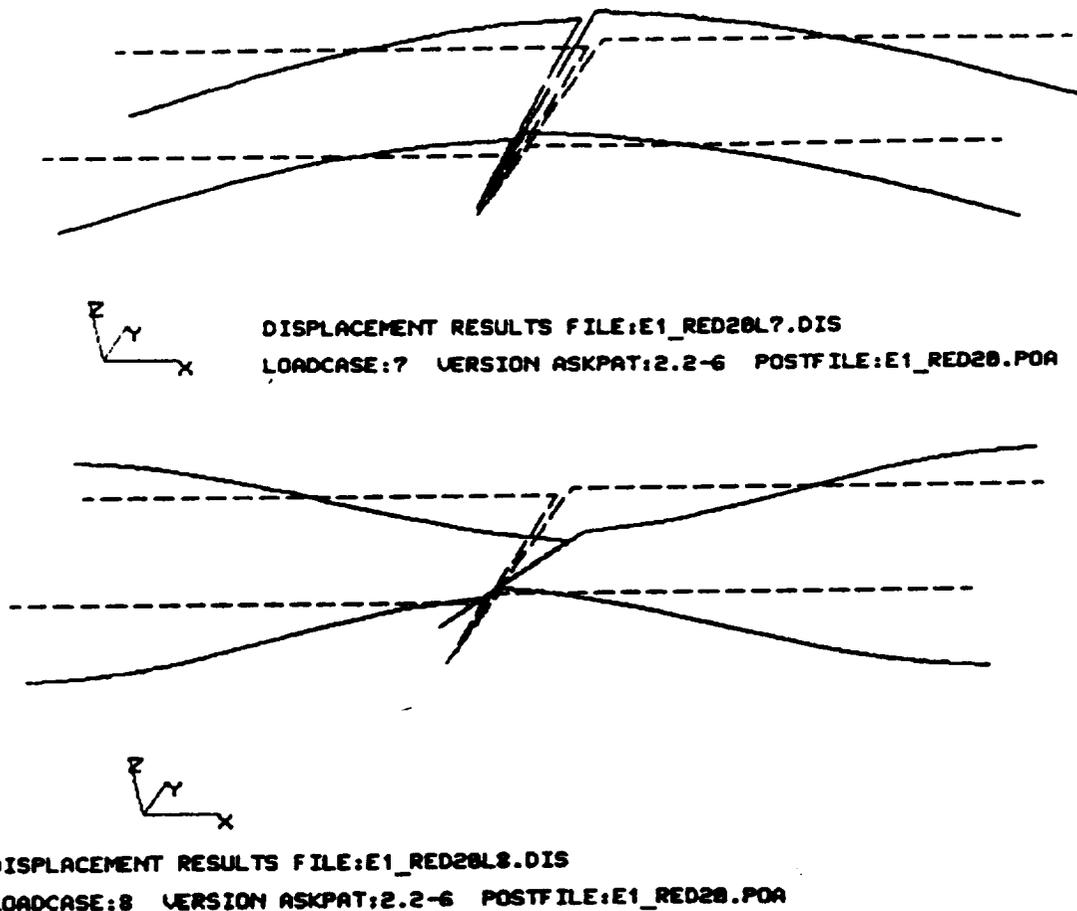
Figure 2: Free-free modes computed with the Craig-Bampton model.
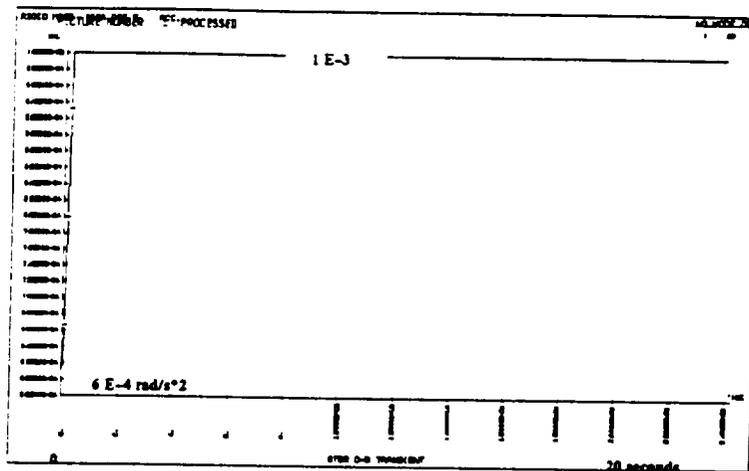
Figure 3: angular acceleration about Y axis at
the HST interface (HST side of the brake)
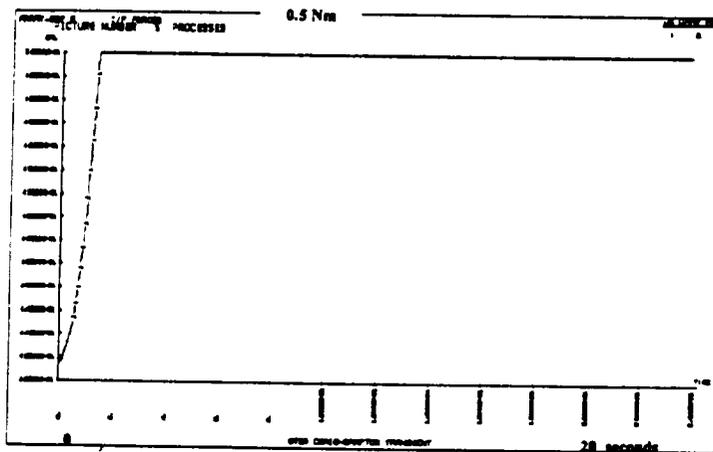
Figure 4: Interface torque about the Y axis
(SA side of brake)
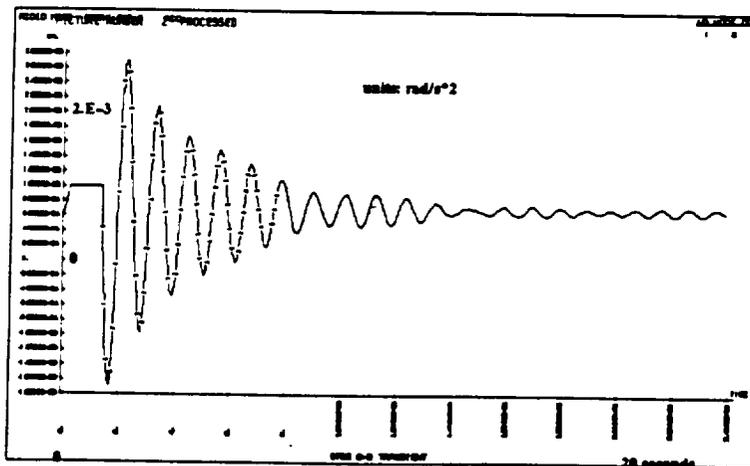
Figure 5: angular acceleration of the SA side of the brake

273
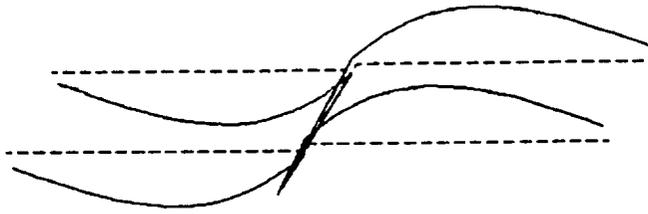
Figure 6:1   Free-free mode 13 of Craig-Bampton
reduced model
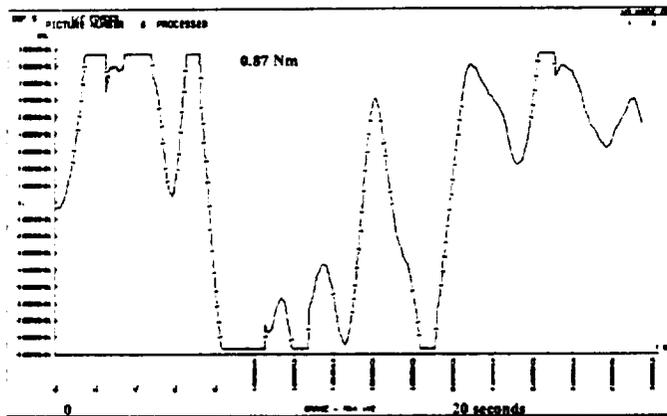
Figure 9: interface torque about Y, SA side of brake
event PO4 wing +V2
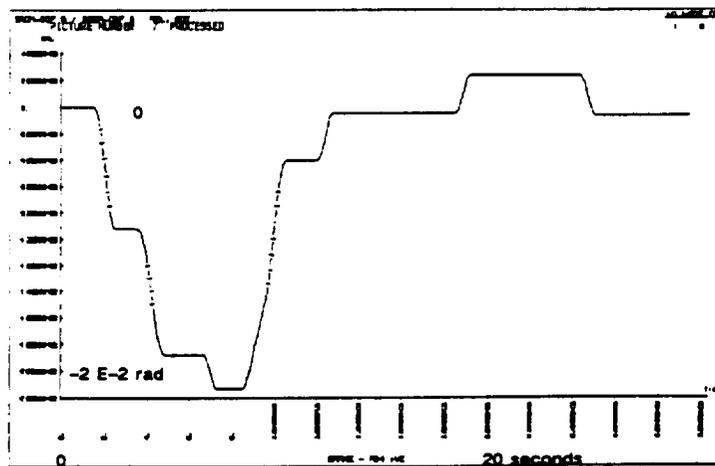
Figure 10: relative rotation around Y of both brake sides

274

CASE	Manoeuvre Type		Infinitely Rigid Brake		Brake allowing slippage (Negative Margins in **Bold**)			
			Bending Moments (Nm)	Margin	Bending Moments (Nm)	Margin	Max. Slippage of the array * (deg)	Percentage of BM reduction
AD11 +V2	RMS Runaway		5.8	–0.10	4.9	0.07	1	16
R12 +V2	RMS Runaway		7.5	–0.30	5.9	**–0.11**	1	21
EP4 +V2	RMS Runaway		6.8	–0.23	3.4	0.54	5	50
EP4 –V2	RMS Runaway		6.1	–0.14	2.6	1.01	3	57
EP6 +V2	RMS Runaway		11.4	–0.54	9.1	**–0.42**	2	20
EP6 –V2	RMS Runaway		10.4	–0.50	8.0	**–0.35**	1	23
PO2 +V2	PRCS Auto hold	Low Z	5.8	–0.10	4.7	0.11	3	19
PO3 +V2	PRCS Auto hold	Low Z	6.1	–0.14	4.0	0.31	1	34
PO4 +V2	PRCS Auto hold	Low Z	8.3	–0.37	5.8	**–0.10**	2	30
PO4 –V2	PRCS Auto hold	Low Z	7.5	–0.30	5.2	0.01	2	31
EP1 +V2	PRCS Auto hold	Nom Z	5.5	–0.05	3.5	0.49	2	36
EP2 +V2	PRCS Auto hold	Nom Z	5.7	–0.08	4.6	0.14	3	19
EP4 +V2	PRCS Auto hold	Nom Z	5.8	–0.10	5.4	**–0.03**	2	7
EP4 –V2	PRCS Auto hold	Nom Z	5.5	–0.05	5.0	0.05	2	9
EP5 +V2	PRCS Auto hold	Nom Z	6.8	–0.23	6.2	**–0.16**	1	9
EP5 –V2	PRCS Auto hold	Nom Z	5.9	–0.12	5.4	**–0.03**	1	8

* Maximum angle obtained during simulation period, 60 seconds.

Table 2: Maximum bending moments in the Solar Array booms.

Note: An modal critical viscous damping ratio of 0.5%for all modes was user for the HST analyses. This value derives from ground test data (ESA–BAe) and agrees with results of in–flight experiment of large solar arrays (Nasa SAFE in–orbit experiment).

Mathematical Theory of a Relaxed Design Problem
in Structural Optimization

Noboru Kikuchi and Katsuyuki Suzuki
Department of Mechanical Engineering and Applied Mechanics
The University of Michigan, Ann Arbor, MI 48109-2215, U.S.A.

Introduction

Various attempt has been made to construct a rigorous mathematical theory of optimization for size, shape and topology (i.e. layout) of an elastic structure. If these are represented by a finite number of parametric functions, as Armand[1,2] described, it is possible to construct an existence theory of the optimum design using compactness argument in a finite dimensional design space or a closed admissible set of a finite dimensional design space, see Pironneau[3] and Buttazzo and Dal Maso[4]. However, if the admissible design set is a subset of non-reflexive Banach space such as $L^\infty(\Omega)$, construction of the existence theory of the optimum design becomes suddenly difficult and requires to extend (i.e. generalize) the design problem to much more wider class of design that is compatible to mechanics of structures in the sense of variational principle. Starting from the study by Cheng and Olhoff[5], Lurie, Cherkaev, and Fedorov[6] introduced a new concept of convergence of design variables in a generalized sense and construct the "G-Closure" theory of an extended (relaxed) optimum design problem. Similar attempt, but independent in large extent, can also be found in Kohn and Strang[7] in which the shape and topology optimization problem is relaxed to allow to use of perforated composites rather than restricting it to usual solid structures. Identical idea is also stated in Murat and Tartar[8] using the notion of the homogenization theory. That is, introducing possibility of *microscale perforation* together with the theory of homogenization, the optimum design problem is relaxed to construct its mathematical theory. It is also noted that this type of relaxed design problem is perfectly match to the variational principle in structural mechanics.

Cheng and Olhoff (1980 - 1982)
  Optimum Design of the Plate Thickness

Lurie, Cherkaev, and Fedorov
  G-Closure Theory, Optimum Composites

Kohn and Strang
  Relaxed Design Problem for a Perforated Material

Murat and Tartar
  Application of the Homogenization Theory to Optimum Design

A Relaxed Design Problem

As briefly described in Introduction, construction of a mathematical theory of structural optimization for distributed parameters (i.e. design variables) requires to extend the design to structures which are perforated by microscale voids. Introducing a perforation characterized by the unit cell shown in the figure, the distributed design variables are defined by the size { $a(x)$, $b(x)$ } of microscale voids and the orientation $\theta(x)$ at an arbitrary point x of a structure. Since x is arbitrary point in the domain Ω of the structure, a, b, and θ are functions in an infinite dimensional function space. That is, since this relaxed design problem is not represented by a finite number of parameters, usual compactness argument is not applicable, but the theory of homogenization is its alternate. A typical optimum design problem is stated by

$$\underset{d}{Minimize} \qquad f(d,u)$$

$$subject\ to$$
$$u \in V : a(u,v)=L(v),\ \forall v \in V_0$$
$$g_u(u) \le g_{u\,max}$$
$$g_d(d) \le g_{d\,max}$$

as shown in Bendsøe and Kikuchi[10], where f is a cost function, d is the design variables, u is the equilibrium displacement, $a(u,v)=L(v)$ is equilibrium of a structure in variational formulation, g are representation of constraints.
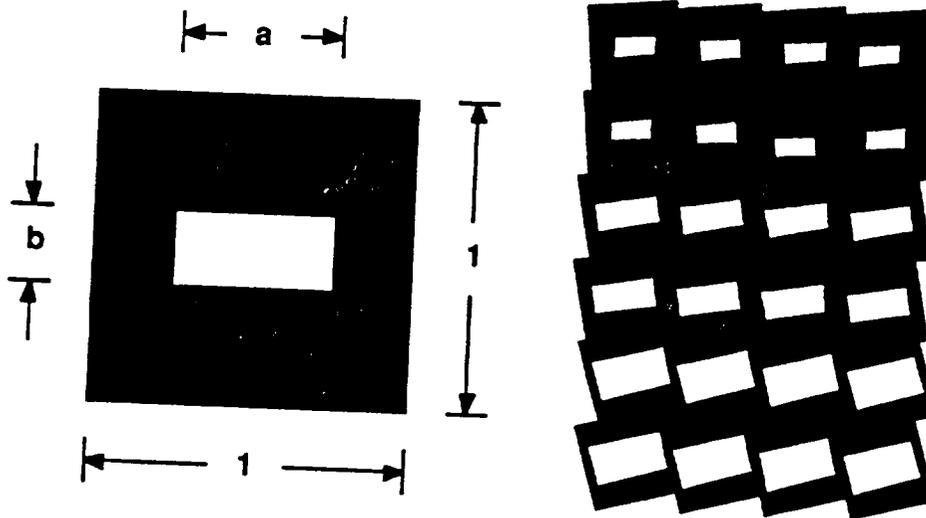
Example - Design of a "Beam" Structure

As simple examples of relaxed design problem, let us consider beam of uniform width with both ends rigid support, and point load at the center. The length of beam is 20 and hight of 1 was taken as design domain.The solution of this problem with thickness as design variable and width fixed is known to be two hinges appear at 1/4 and 3/4 of length.

The objective function was taken as mean compliance of structure, i.e. load times displacement at loading points. Using optimality criteria method in relaxed design problem described before, and volume constraint 10, 15, 18 with total volume of design area 20, following results are obtained. These results clearly show hinges at 1/4 and 3/4 of length points in each case. Also it is noteworthy that when constraint on volume is small, the beam becomes sandwich type beam, i.e. upper and lower surface and weak core inside. Although this type of structures are known to be preferable when bending is applied,and used often in aerospace industry, traditional method with thickness as design variable cannot generate them unless this type of structure is assumed a priori.

A Design Problem of a Beam Structure
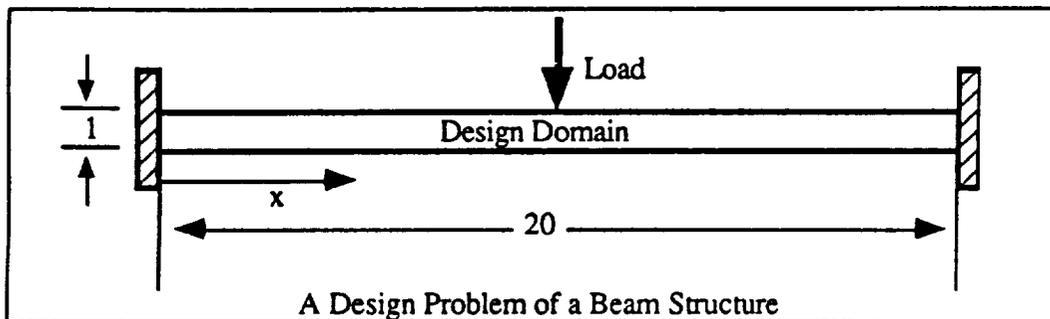
Volume Constraint 10

Volume Constraint 15

Volume Constraint 18

Optimal Layout of a Beam like Structure

Example - Design of a "Truss" Structure

Second example is bridge type structure. The structure is supported in both end, and hight of bridge is limited to one half of the length of bridge. Point load is applied at the middle points of two support points. Volume constraint is taken 25, 50, 75 respectively while total volume of design area is 200. The results using relaxed design problem is shown below. Each result consist from two types of structure, a bridge that connect both supports and tension member that connect bridge and loading points. When constraint on volume is small (25), the design clearly shows 9 bar truss like structure, although the bridge part generate small holes inside since it is not totally free from bending effect.

279
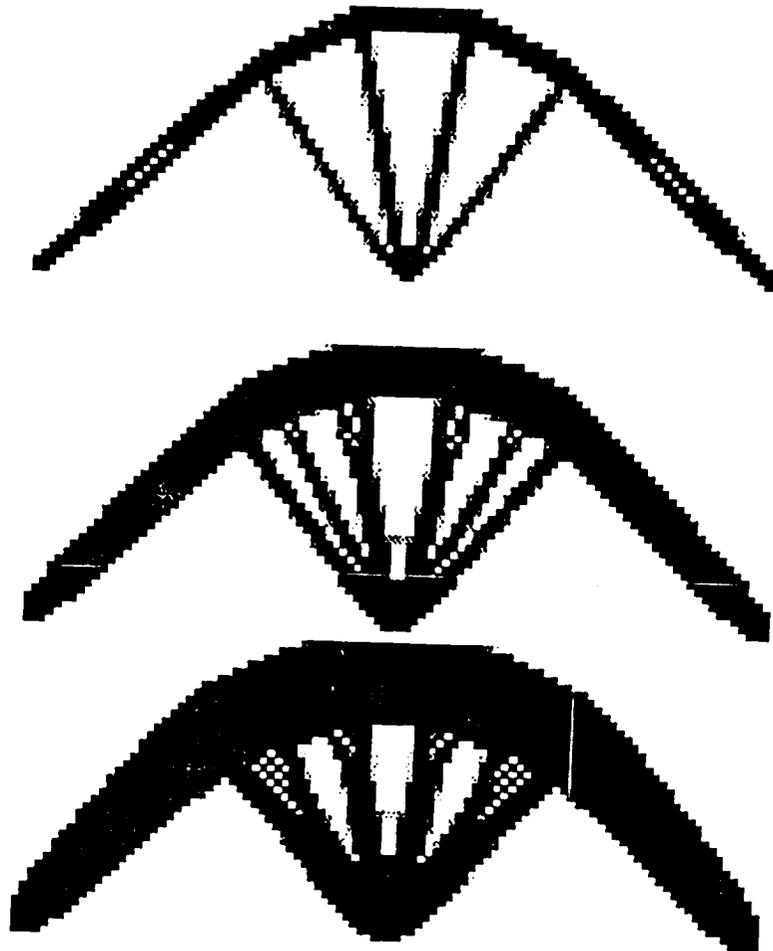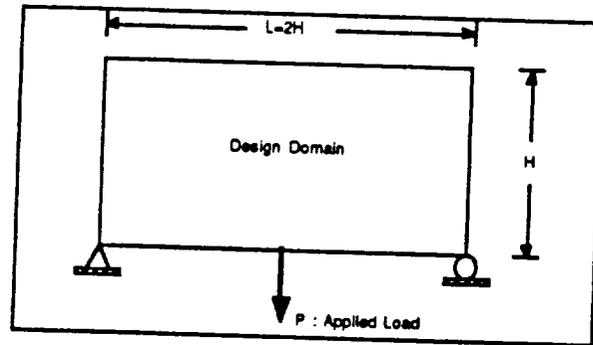
Example - Optimum Reinforcement of a Plate

As a examples of application of relaxed design problem to plate, design of simply supported square plate subject to point load at the center is considered. The thickness of support layer and thickness of stiffener are fixed. Relaxed design problem of minimize mean compliance subject to volume constraint is solved using optimality criteria method.

The results are shown below. In cases volume constraint is large, the final design clearly shows four hinge lines of 45 degree inclined lines. It is reminded that two hinges are generated for a beam case also. Thus, hinges in the optimal reinforcement is expected one. It should be noted that this does not mean discontinuity of the transverse displacement along these hinge lines. But only the slope is discontinuous, and it is still admissible in the variational formulation of an elastic plate defined in the Sobolev space $H^2(\Omega)$.

In cases volume constraint is relatively small, cross shape support with with diagonal support is optimal. It is noteworthy that hinges also appear in diagonal supports, although they are not so conspicuous as in large material case. That cross shape supports become thinner as they approach edges can be explained that minimum plate will disperse the load as you approaches the edges.
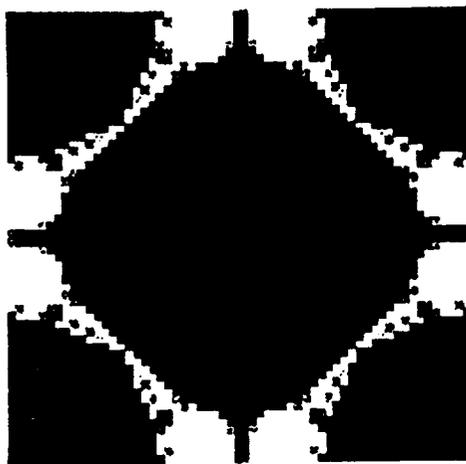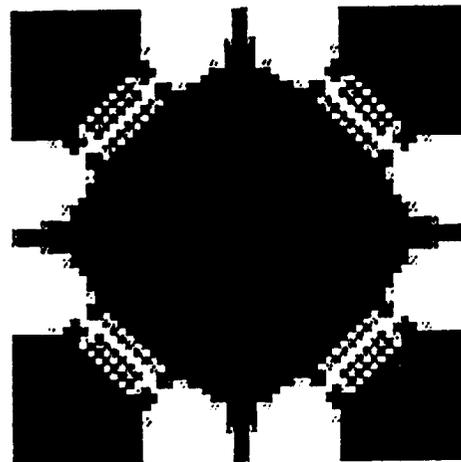
(a) volume 720/900 (b) volume 630/900
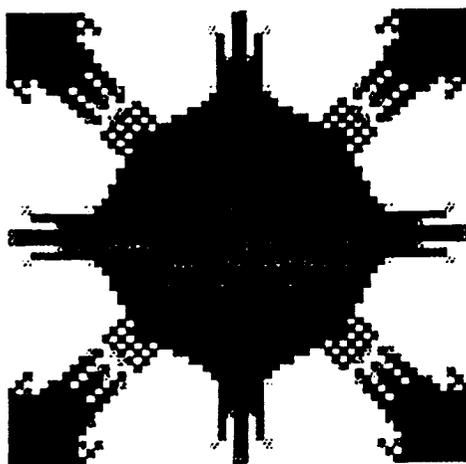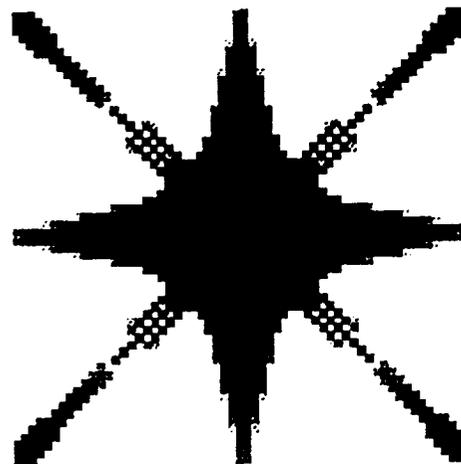
(c) volume 450/900 (d) volume 270/900

Optimal Layout of a Plate with Point Load

280

Example - Optimum Reinforcement of a Shell

As a comparison to previous example, same type of design problem is employed, while in previous case there was no curvature in shell and there is only bending effect, but in this case there is curvature and coupling of bending and in-plane effect exists. The shape of shell is $z = z_{max} \sin(\pi x / x_{max}) \sin(\pi y / y_{max})$ with $z_{max} / x_{max} = z_{max} / x_{max} = 1/12$. Again, boundary condition is simply support and load is applied at the center of square.

The results shown below are quite different from previous design of plate. Surrounding area is stiffened, and loading point and surrounding stiffener is connected by diagonal support.

These quite different results from plate case is very natural consequence considering the fact that in previous plate case, only bending moment is applied since bending and tension are totally decoupled, while in this shell with curvature, since there is coupling between bending and tension and rather tension is dominant, similar results to two dimensional problem appears.
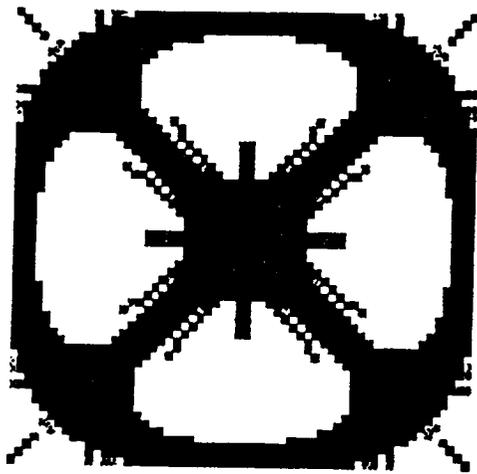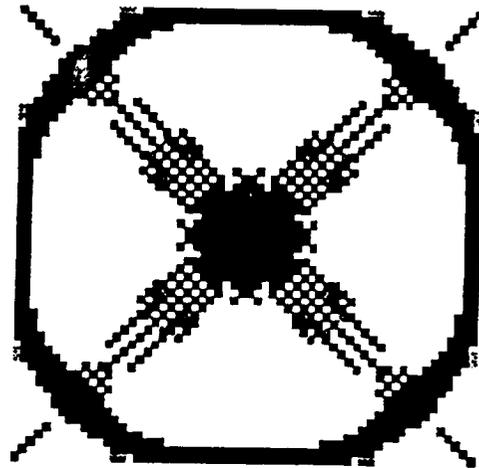
(a) volume 450/900   (b) volume 270/900

Shell with Point Load at the Center

Example - A Three-Dimensional Structure

The relaxed design problem can be applicable to 3 dimensional solid. In this case, design variables are 6, 3 for sizes of microscopic rectangle body holes and 3 for rotational angles. Same type of formulation and numerical method as before, i.e. minimize mean compliance with constraint on vulume, and discretized using finite element method and optimized using optimality criteria method.

As a example, the cantilever rectangle plate subjected to shear force is solved. The size of plate is 32x20 with thickness 4, and 4 elements were used in thickness direction. One end is clamped and the other end is loaded at the center point of section. Volume constraint was set 800, while volume of total design domain is 2560.

The result are shown below as each 4 layer in thickness direction, from top to bottom. Top and bottom layer has triangular shape of full material area to support bending moment and two layers in between have very thin rim to support interaction of top and bottom layers. This types of structure is known as "Sandwitch structure", and supposed to be one of the best structure for bending.

1st Layer

2nd Layer

3rd Layer

4th Layer

282
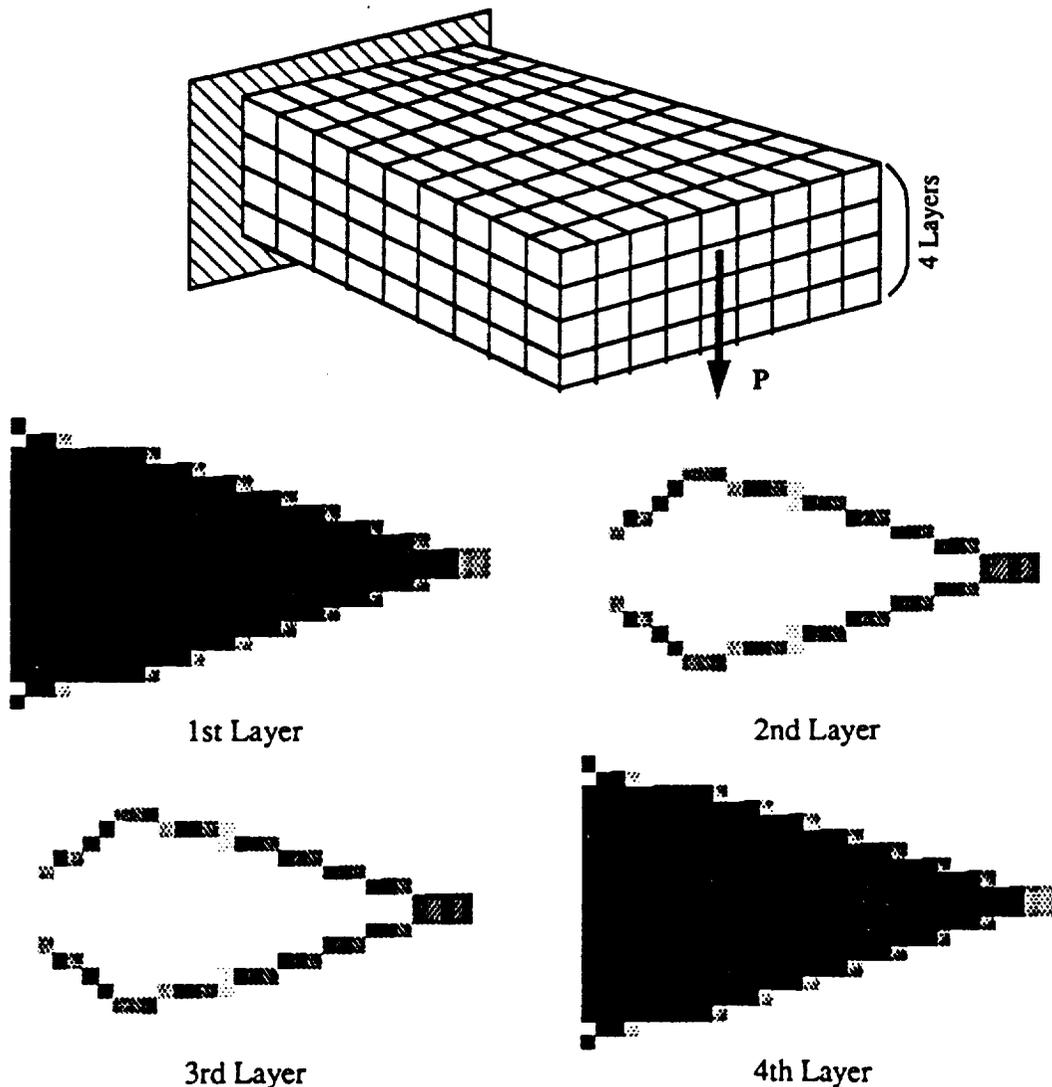
Convergence of Finite Element Methods

The natural question arises is whether the shape and topology, i.e., the configuration of the structure obtained as the optimal design converges to the unique one as finite element meshes are uniformly refined, while other conditions are fixed. For this purpose, the relaxed design problem described below is solved with several different sizes of finite element meshes, 32x20, 48x30, 64x40, 80x50, and rectangular design domain with boundary condition left end fixed and load being applied on right side.

The results shown below clearly shows convergence to certain topology as mesh becomes finer and finer. Therefore, it is natural to consider that as mesh size becomes zero, the results converge to the final configuration that is design variable as continuous function, and the topology of the design with continuous design variable is almost same as that of finite mesh size.

Design Domain for Bending of a Short Cantilever
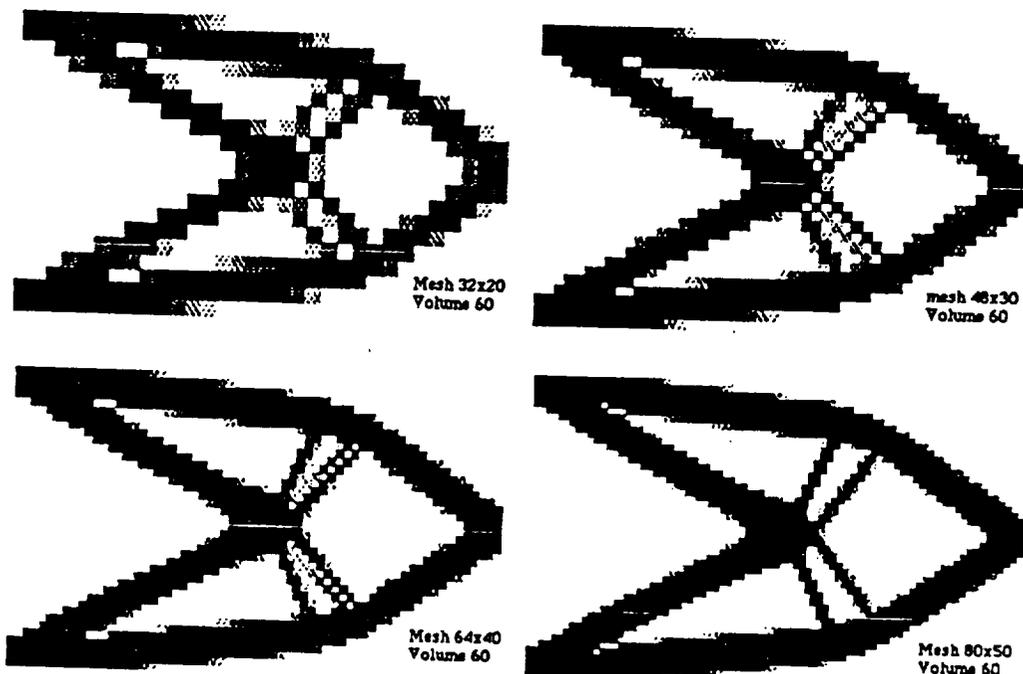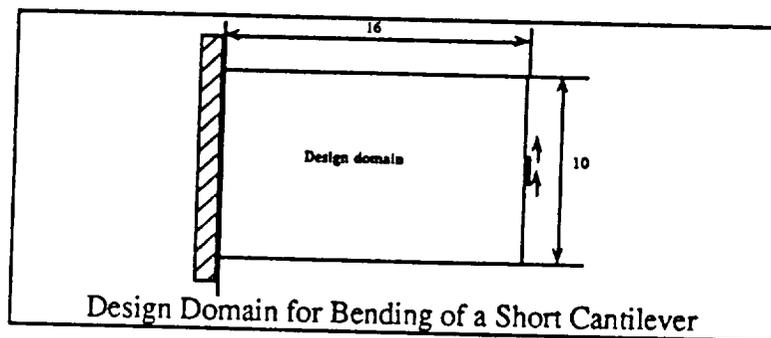
Convergence of the Optimal Configuration ($\Omega_s = 60$)

References

[1] Armand, J.L., Lodier, B . Optimal Design of Bending Elements. Int. J. Num. Meths. Engrg 13, 373-384 (1978).

[2] Armand, J.L., Lurie, K.A. and Chercaev, A.V. . Optimal Control Theory and Structural Design. New Directions in Optimum Structural Design,John Wiley & Sons (1984) 211-229.

[3] Pironneau, O . Optimal Shape Design for Elliptic Systems. Springer Verag, NewYork,1986.

[4] Buttazzo, G and Dal Maso, G . Shape Optimization for Dirichlet Problems: Relaxed Solutions and Optimality Conditions. Workshop on Composite Media and Homogenization Theory, (1990) Trieste, Italy.

[5] Cheng, K.T. and Olhoff, N. . An investigation concerning optimal design of solid elastic plates. Int. J. Solids and Structures 17 (1981) 305-323.

[6] Lurie, K.A., Cherkaev, A.V., and Fedorov, A.V. . Regularization of optimal design problems for bars and plates, Parts I and II. J. Optim. Theory Appl. 37-4 (1982) 499-521, 523-543.

[7] Kohn, R. and Strang, G. . Optimal Design and relaxation of variational problems, Parts I, II, and III. Communications on Pure and Applied Mathematics, XXXIX (1986) 113-137, 139-182, 353-378.

[8] Murat, F. and Tartar, L. . Optimality conditions and homogenization. in Nonlinear Variational Problems, Pitman Advanced Publishing Program, Boston, 1985, p1-8.

INTEGRATED TOPOLOGY AND SHAPE OPTIMIZATION IN STRUCTURAL DESIGN

by

M. Bremicker, M. Chirehdast,
N. Kikuchi and P. Y. Papalambros

Department of Mechanical Engineering and Applied Mechanics
The University of Michigan, Ann Arbor

ABSTRACT

Structural optimization procedures usually start from a given design topology and vary its proportions or boundary shapes to achieve optimality under various constraints. Two different categories of structural optimization are distinguished in the literature, namely sizing and shape optimization. A major restriction in both cases is that the design topology is considered fixed and given. Questions concerning the *general layout* of a design (such as whether a truss or a solid structure should be used) as well as more detailed *topology features* (e.g., the number and connectivities of bars in a truss or the number of holes in a solid) have to be resolved by design experience *before* formulating the structural optimization model. Design quality of an optimized structure still depends strongly on engineering intuition. This article presents a novel approach for initiating formal structural optimization at an earlier stage, where the design topology is rigorously generated in addition to selecting shape and size dimensions. A three-phase design process is discussed: an optimal initial topology is created by a homogenization method as a gray level image, which is then transformed to a realizable design using computer vision techniques; this design is then parameterized and treated in detail by sizing and shape optimization. A fully automated process is described for trusses. Optimization of two-dimensional solid structures is also discussed. Several application-oriented examples illustrate the usefulness of the proposed methodology.

Topology Optimization (Phase I)

Recently, Kikuchi and Bendsøe [1] presented a homogenization method for solving generalized structural layout problems. An initial domain, provided by the designer, is discretized into finite elements. Boundary conditions are also supplied at this stage. The density and stiffness properties of the elements are determined by applying homogenization to the model of a unit cell with a rectangular hole; see Figure 1. Using the dimensions (a and b) and orientation angle (θ) of the holes as design variables, the method searches for the minimum compliance or maximum stiffness of the structure subject to a volume constraint. In other words, homogenization solves the problem formulated in Eq. 1. A given amount of material is thus redistributed with microstructure properties as variables. The resulting material distribution corresponds to an optimum topology that can vary from truss-like configurations to closed solid shapes. The method has been successfully implemented to date for two-dimensional structural problems [1,2].
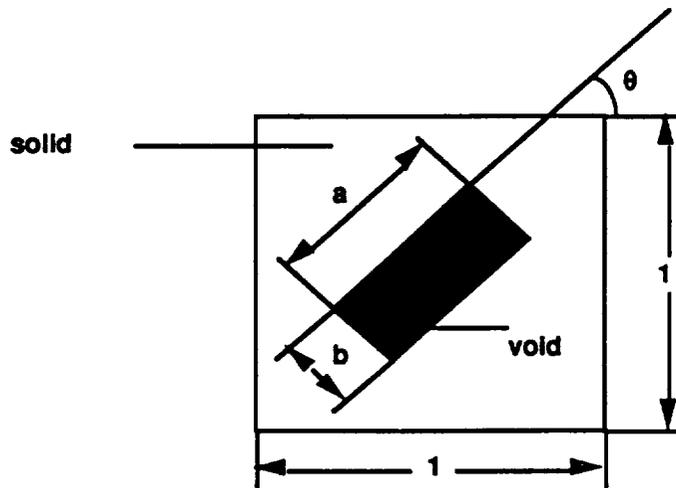
Figure 1. Definition of a microscale hole, i.e., a finite element for homogenization method.

Minimize   Mean compliance of structure
subject to

Equilibrium equations   (1)

Amount of material as a percentage of total domain volume

286

Three-Phase Design Methodology (ISOS)

The integration of a topology-generation step into an overall design optimization process can be performed according to the following general scheme:

Phase I: Generate information about the optimum topology for the structure.
Phase II: Process and interpret the topology information.
Phase III: Create a parametric model for detailed shape and size optimization of the derived topology and apply standard optimization techniques.

This scheme is the basis for an Integrated Structural Optimization System (ISOS), outlined in Fig. 2, and described in general terms by Papalambros and Chirehdast in [3].
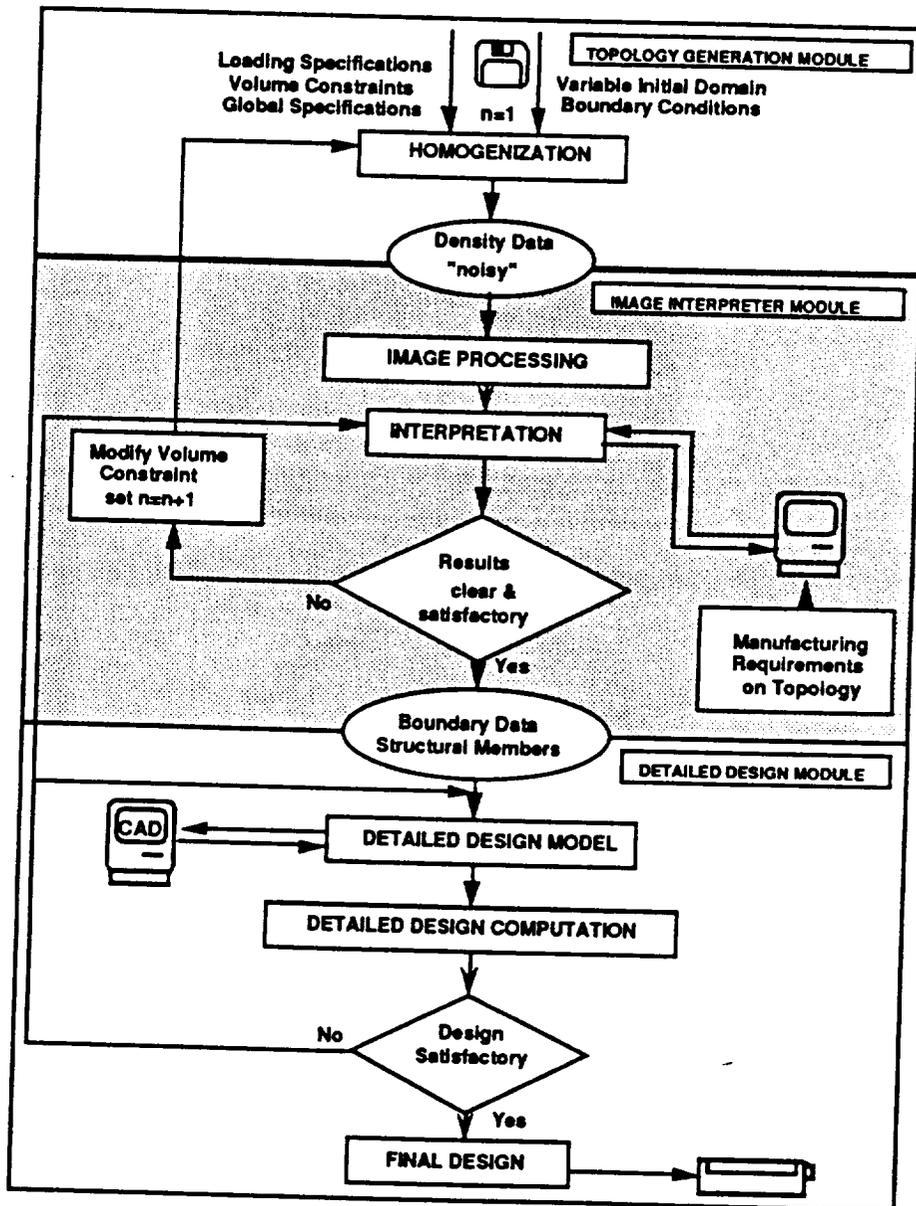
Figure 2. Basic flow chart for the three-phase structural optimization system ISOS.

287

Example 1

The operations performed by ISOS in Fig. 2 are best illustrated through the simple example shown in Fig. 3. Figure 3(a) is the starting point. Figure 3(b) is the output of Phase I and input to Phase II. Figure 3(c) is the output of Phase II, and is converted by the designer to Fig. 3(d) as input to Phase III.
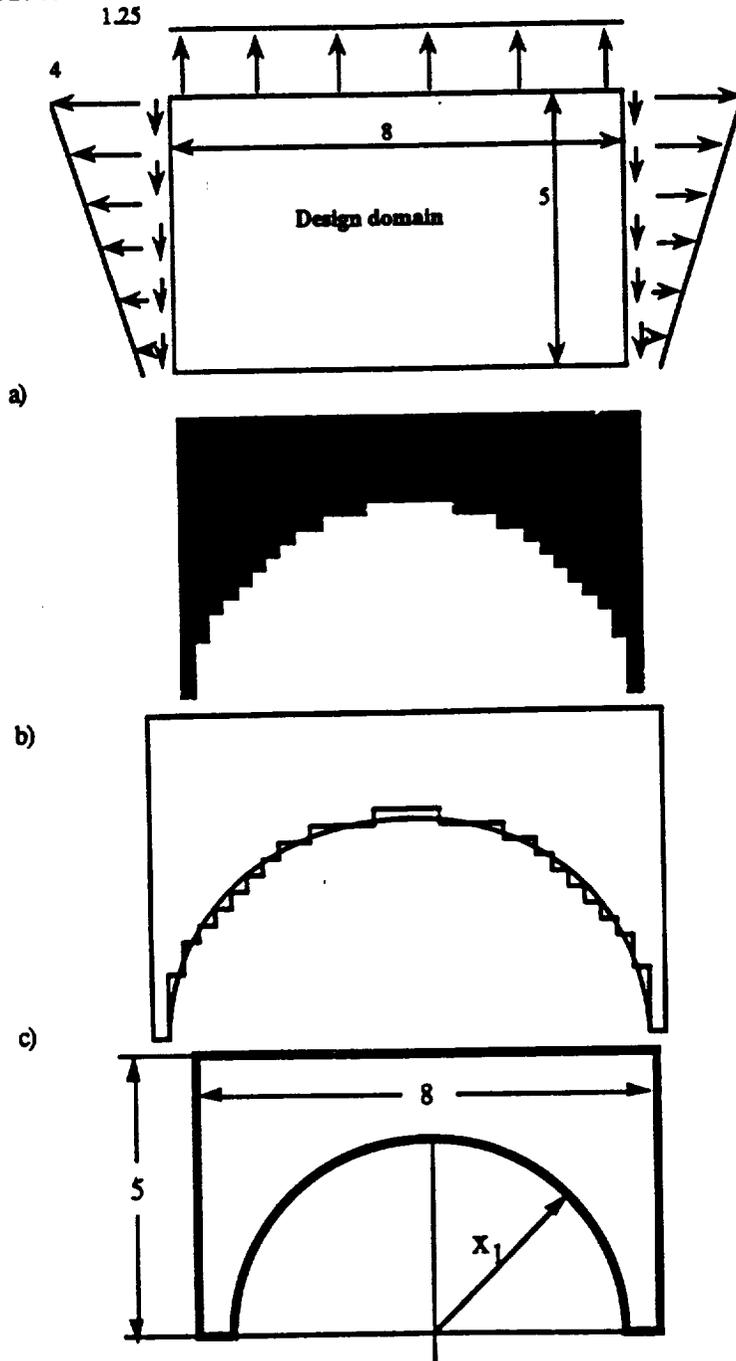
Figure 3. Design example 1: a) design domain and boundary conditions; b) density distribution generated by homogenization; c) processed image; d) user interpretation and detailed design model (design variable x_1).

Image Interpretation Module (Phase II)

The general image interpretation module (Phase II) is depicted in Fig. 4 in more detail. The module is a representation manipulator, its basic purpose being extraction and processing of higher level knowledge from the density arrays generated by homogenization using image processing and computer vision techniques. Another purpose of Phase II is to impose nonstructural requirements on topology, such as manufacturing constraints, which is accomplished using domain-dependent rule-based systems. Currently, homogenization can only handle two-dimensional problems and therefore three different types of two-dimensional structures need to be treated by Phase II; these are plane stress/plane strain structures, trusses, and frames. In [3] and [4] techniques for dealing with plane stress/plane strain structures and trusses are discussed respectively. The degree of automation for treating trusses in Phase II is higher than that for solid structures. Investigation of frames will be the focus of our research in the near future.
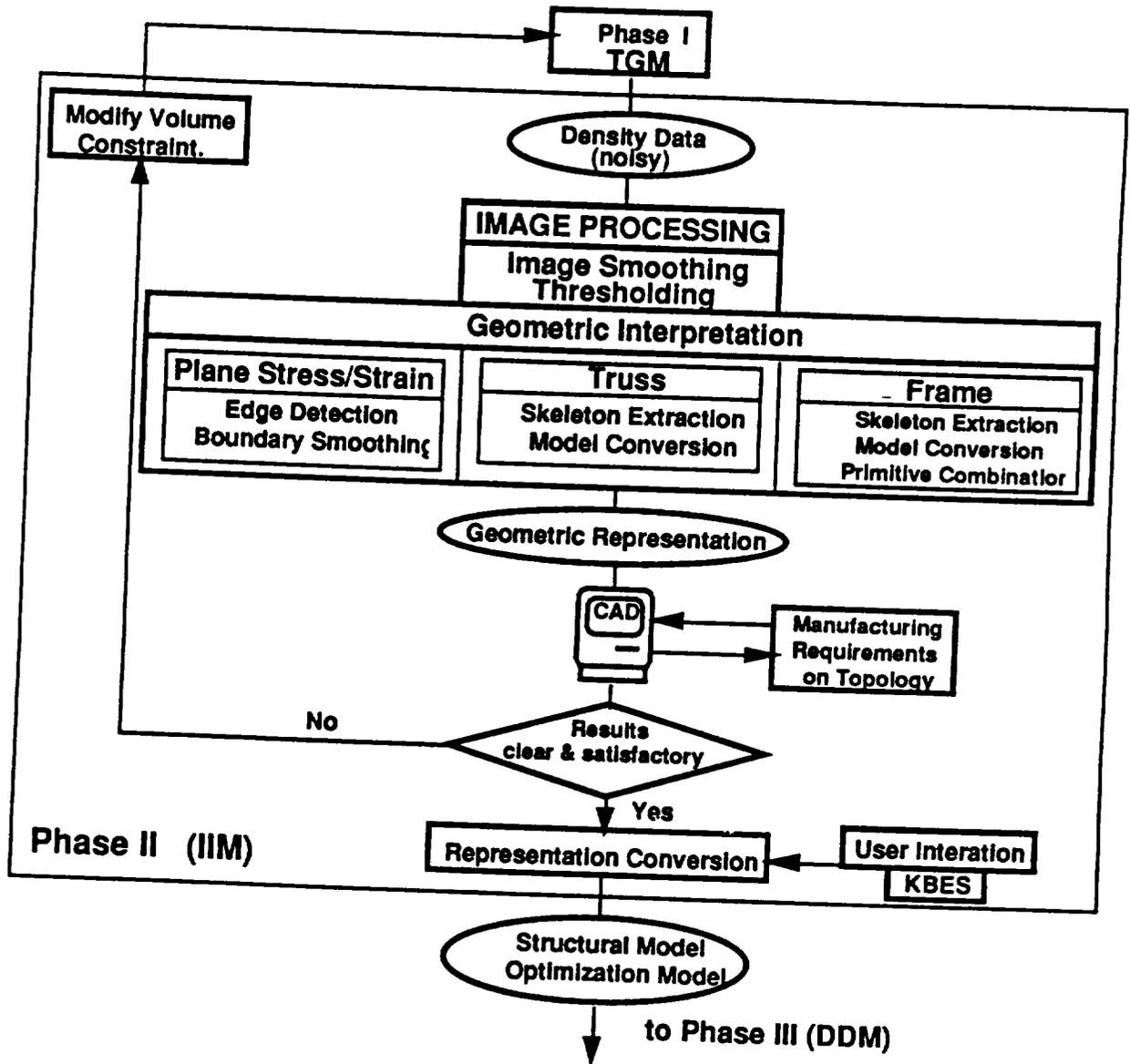
Figure 4. Phase II (Image Interpreter Module) in more detail.

289

Detailed Design Module (Phase III)

A detailed design model is set up in Phase III to perform sizing and shape optimization. The number of design variables and constraints is usually moderate at this point, and general mathematical programming algorithms can be used, allowing versatility in objective and constraint function specifications. In ISOS, an evolved version of the structural optimization package SAPOP [5,6] is used; it consists of a finite element program, several constrained optimization algorithms and standard pre- and post-processors, including automatic mesh generation. At this stage, the designer can once again check if the proposed design is satisfactory. Figure 5 depicts a schematic overview of SAPOP and Phase III. Two distinct models of the design are required by SAPOP, namely structural and optimization model. For further detail on this module refer to either SAPOP references [5,6] or previous publications on ISOS [3,4].
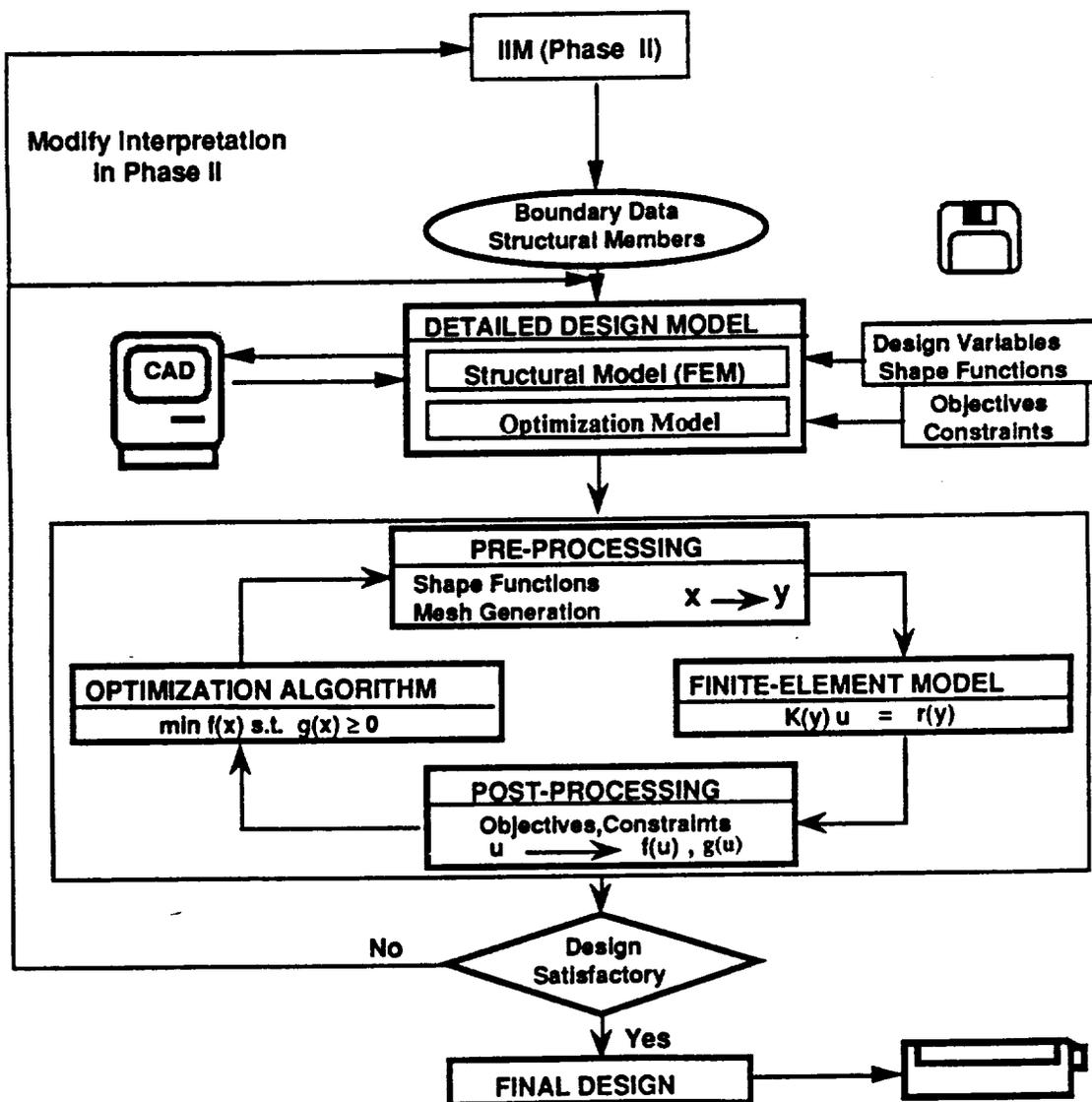
Figure 5. Detailed flow chart of Phase III (Detailed Design Module).

Example 2 - Truss Optimization (Phase I)

A cantilever structure is to be designed for minimum weight supported and loaded as shown in Fig. 6(a). Maximum allowable stress limit is set at 180 N/mm^2 and the maximum displacement at the loading point is limited to 10 mm. The initial design domain is prescribed to be a rectangle, as shown in Fig. 6(a). The homogenization procedure is applied using the model of Eq. (1) and disregarding explicit stress and deflection constraints. The design domain is discretized into 40 by 60 finite elements. Since each finite element represents a unit cell and the shape of each unit cell is defined by three design variables, the homogenization method has to solve an optimization problem with 7200 design variables. Figure 6(b) shows the optimum material distribution for this problem for a solid-to-void ratio of 1/3. The solution requires about 7 hours of computational time on an Apollo 4000 workstation.

(a)

(b)
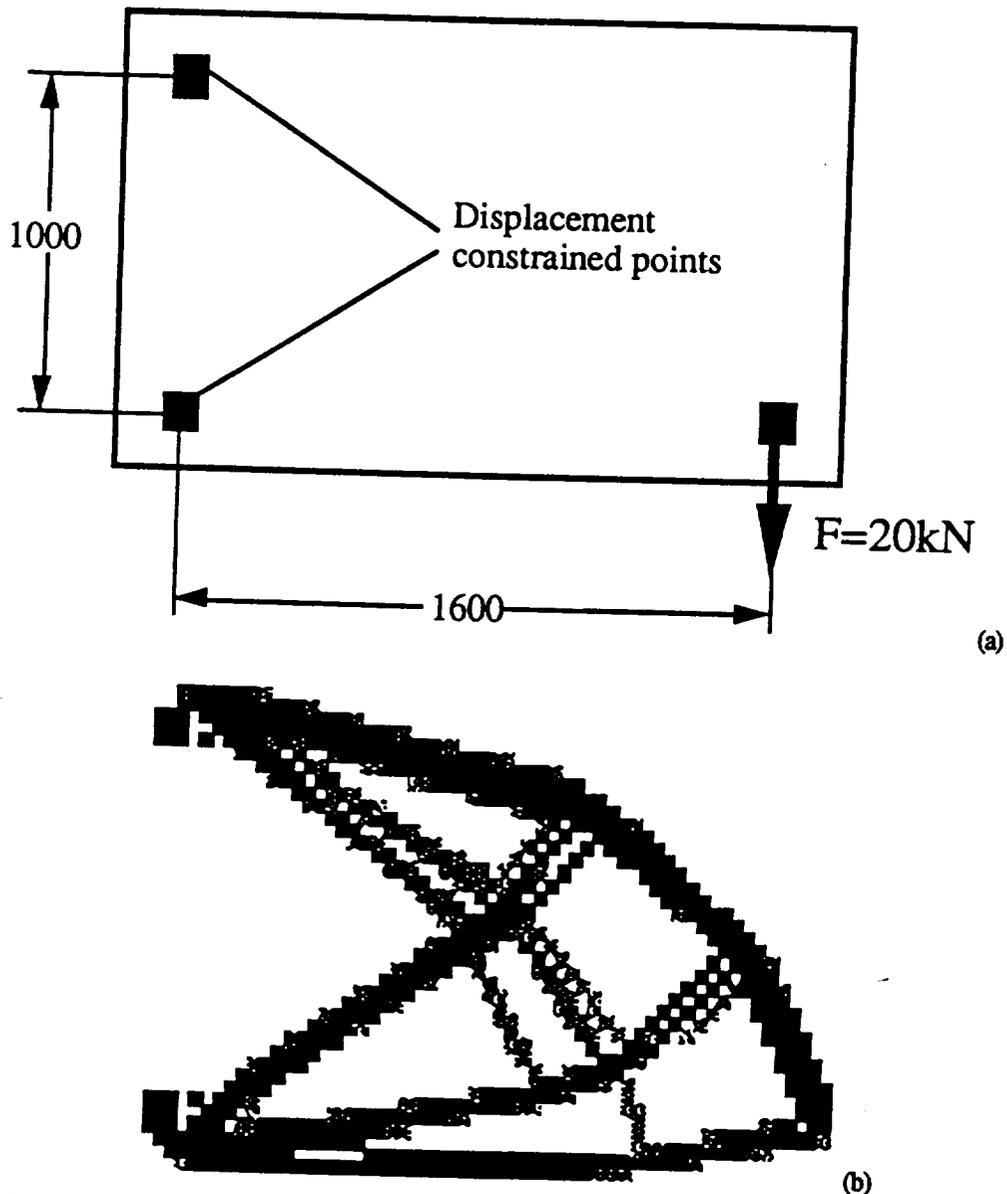
Figure 6. (a) Design specifications and (b) Optimum material distribution for Example 2.

Example 2 (Output of Phase II and Some Alternative Designs)

The final output of Phase II for this example is shown in Figure 7 which results automatically by performing the steps explained in [4]. At this stage, we search for the minimum weight design that satisfies the above specified stress and displacement constraints. Design variables are cross sections of truss members and the coordinates of the unrestrained and unloaded nodes. To study the optimality of the proposed topology, it is compared with some common truss topologies. Truss models of four different topologies were chosen to conduct the study and are shown in Fig. 8, dotted lines indicating truss elements removed by sizing optimization.
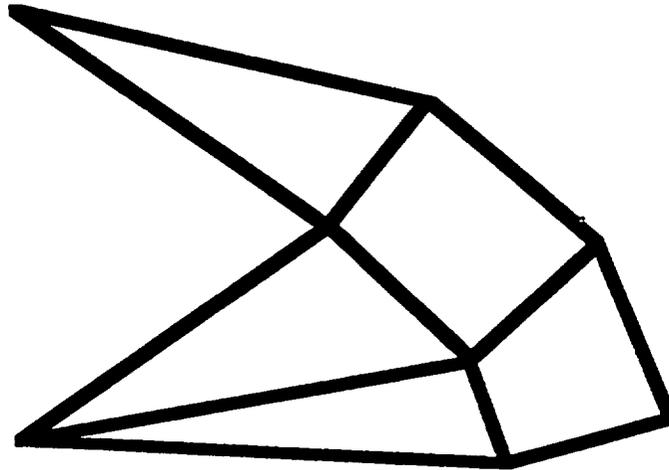
Figure 7. Output of Phase II for Example 2.

Five-Bar Topology

Ten-Bar Topology

Homogenization Topology

Fourteen-Bar Topology
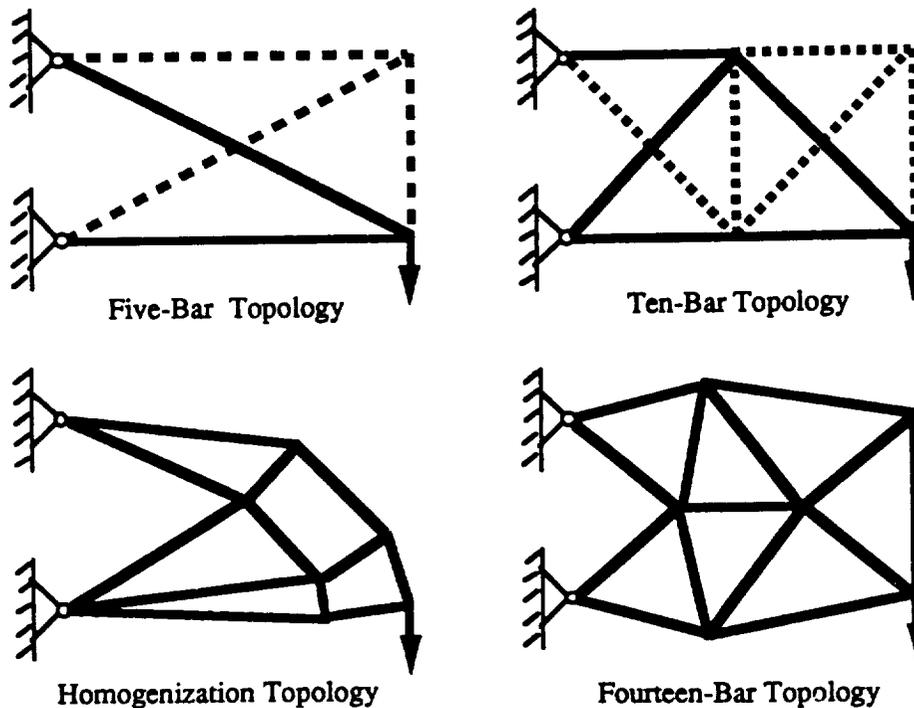
Figure 8. Alternative engineering solutions for the bracket problem.

Example 2 (Phase III and Final Results)

Table 1 shows the weights achieved by sizing optimization where design variables are only cross-sections of the bars. In Table 2, the results for a combined sizing and geometry optimization are given where additionally coordinates of unrestrained and unloaded nodes are introduced as design variables. The topology proposed by Phase I is clearly leading to the lowest weight. In the case of combined sizing and topology optimization, the positions of the nodes of the homogenization truss move only slightly, in contrast to the nodes of the other truss designs. The nodes of the homogenization truss are nearly optimally located even before geometry optimization is performed in Phase III.

Table 1. Comparison of initial and optimal weights resulting from sizing optimization.

Model	Weight [kg]	
	Initial	Final
Five-Bar Truss	31.29	28.012
Ten-Bar Truss	40.51	25.527
Homogenization Truss	32.27	23.294
Fourteen-Bar Truss	42.51	28.095

Table 2. Comparison of initial and optimal weights resulting from sizing and geometry optimization.

Model	Weight [kg]	
	Initial	Final
Five-Bar Truss	31.29	28.012
Ten-Bar Truss	40.51	23.714
Homogenization Truss	32.27	22.848
Fourteen-Bar Truss	42.51	23.540

Example 3 - Solid Two-Dimensional Structure (Phase I)

A bracket is loaded by a single force and a moment, shown in Fig. 9. A minimum weight design subject to stress and displacement constraints is sought. The allowable equivalent stress is 50 N/mm² and the allowable deflection at the loading point is 0.1 mm. The homogenization output is shown in Fig. 10 where the imposed volume constraint is 50%.
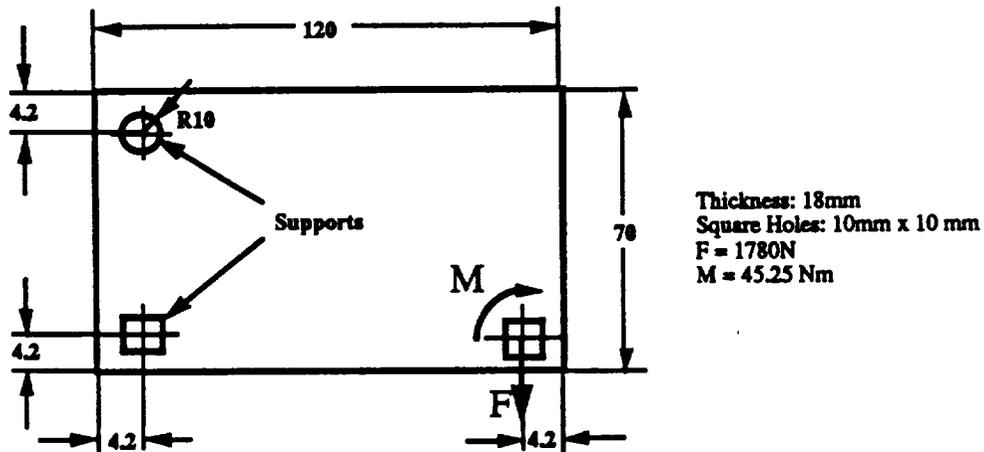
Figure 9. Boundary specifications and initial design domain for Example 3.
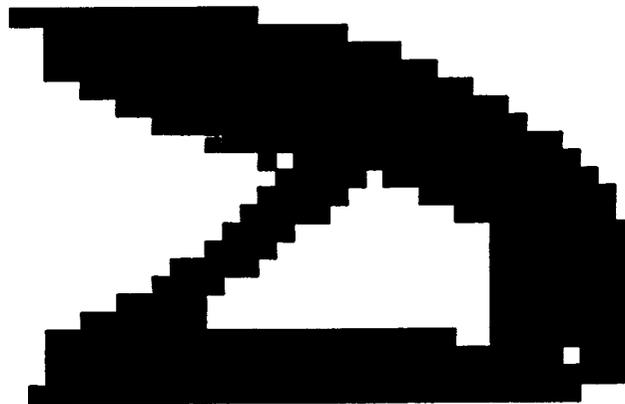
Figure 10. Optimum material distribution for Example 3 (solid/void ratio=1/1)

Example 3 (Phases II and III)

The image of Fig. 10 is interpreted as a two-dimensional solid and subsequently filtered and smoothed by the image processing tools described in [3]. Figure 11(a) shows the detected non-smooth edges. In Fig. 11(b) the smoothed boundaries are depicted. Note that linear functions as well as B-splines are used for smoothing the boundaries. Based on these images, an initial design is constructed. To achieve structural optimality while accounting also for manufacturing cost, only straight lines and fillets are used for boundary representation. The finite element model of the resulting design as an input for Phase III is shown in Fig. 11(c). Note that the transition from the image shown in Fig. 11(b) to the model of Fig. 11(c) is performed manually. Automation of interface between Phases II and III for plane stress/strain structures is under development. The detailed design model has weight as the objective with stress and displacement constraints as defined above, and allows variations in the positions of the fillets in the design. Representation of the fillets is discussed in [4] in more detail. This particular detailed design model is only one possibility among many others, e.g., one could define nonlinear shape functions to represent the boundaries, provided the manufacturing requirements would permit the designer to do so. The detailed optimum design is presented in Fig. 11(d), achieved after 12 optimization iterations with a sequential quadratic programming method, and represents the final minimum weight design satisfying both structural and manufacturing constraints.

(a)

(b)

(c)

(d)
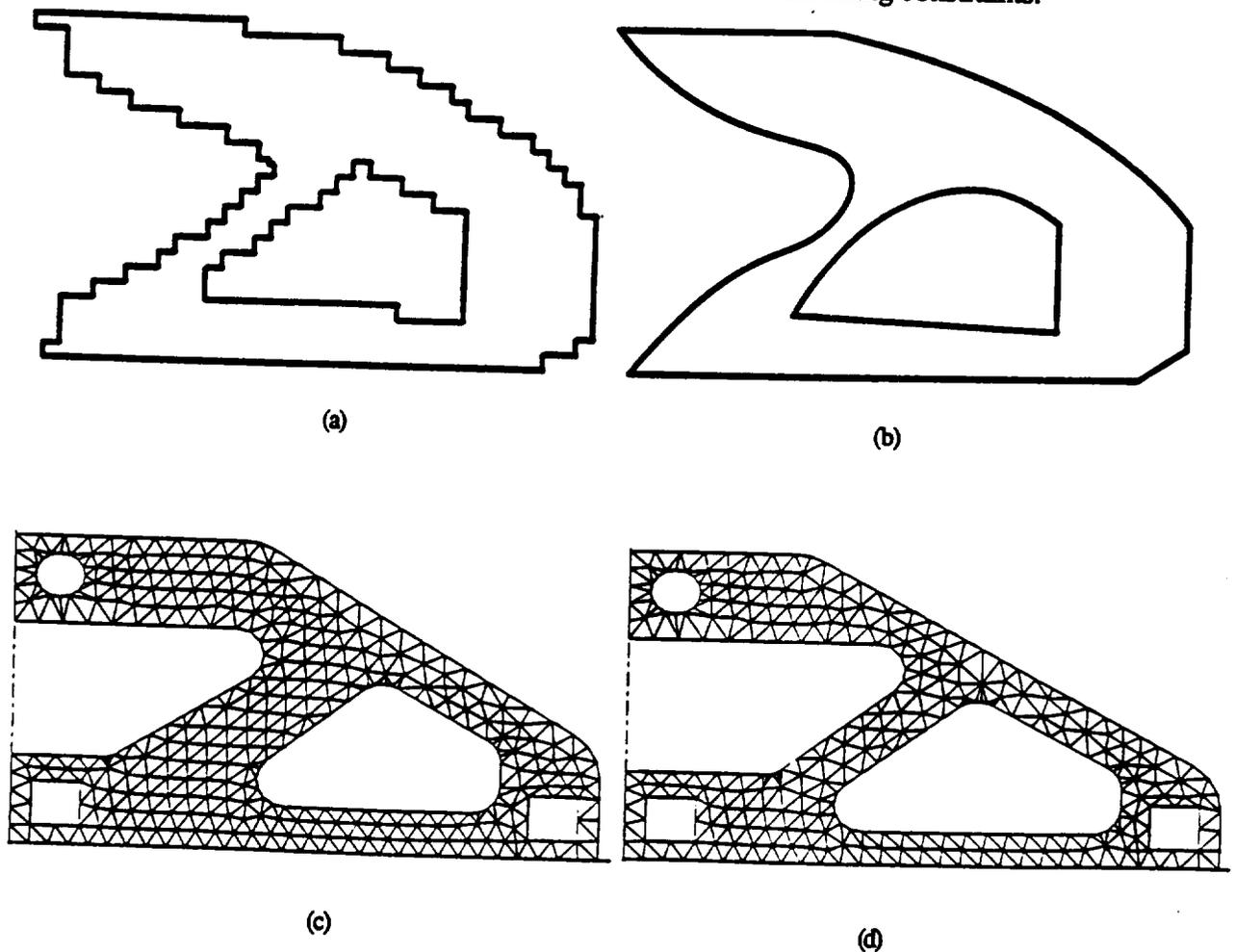
Figure 11 (a)Non-smooth and (b) smooth boundary representation of image shown in Fig. 10; finite-element models of (c) initial and (d) final bracket design.

Conclusions

The integrated structural optimization system (ISOS) described here offers an exciting new capability for structural design. The ability to generate topologies on a rigorous analytical foundation opens the way for integrating several tools from different disciplines such as structural mechanics, manufacturing, computer vision, expert systems, and mathematical optimization. A unique attribute of this system is its capability to allow examination of design constraints in many different domains. The integrated program ISOS is obviously in an evolving state. Modules I and III are sufficiently developed to operate in an automated way. In Module II, the image processing capability is fairly well developed, but the image interpretation module is not yet automated and will provide a continuing challenge. It should be noted that progress offered by other researchers in feature-based design and design for manufacturability can directly benefit automation efforts for this module.

Acknowledgements

References

[1] Bendsoe, M., Kikuchi, N., Generating Optimal Topologies in Structural Design Using a Homogenization Method. *Computer Methods in Applied Mechanics and Engineering,* 197-224 (1988).

[2] Suzuki, K., Kikuchi, N., Shape and topology optimization for generalized layout problems using the homogenization method, *Computer Methods in Applied Mechanics and Engineering,* to appear, (1990).

[3] Papalambros, P., Chirehdast, M., An Integrated Environment for Structural Configuration Design, *Journal of Engineering Design,* 1(1), (1990).

[4] Bremicker, M., et al., Integrated Topology and Shape Optimization in Structural Design. *Tech. Report UM-DL-90-01,* Department of Mechanical Engineering and Applied Mechanics, The University of Michigan, Ann Arbor, (1990).

[5] Eschenauer, H., Post, P.U. and Bremicker. M., Einsatz der Optimierungsprozedur SAPOP zur Auslegung von Bauteilkomponenten. *Bauingenieur,* 63: 515-526 (1988).

[6] Bremicker,M., et al., Optimization Procedure SAPOP - A General Tool for Multicriteria Structural Design. in: Eschenauer, H., Koski, J., Osyczka, A., (Eds.) *Multicriteria Design Optimization.* Berlin, Springer-Verlag, May (1990).

SHAPE OPTIMIZATION FOR MAXIMUM STABILITY

AND DYNAMIC STIFFNESS

W. SZYSZKOWSKI

Department of Mechanical Engineering
University of Saskatchewan, Saskatoon, Canada

1 INTRODUCTION

Any optimization of structures for maximum stability or for maximum dynamic stiffness deals with an eigenvalue problem. The goal of this optimization is to raise the lowest eigenvalue (or eigenvalues) of the problem to its highest (optimal) level at a constant volume of the structure. Likely the lowest eigenvalue may be either inherently multimodal or it can become multimodal as a result of the optimization process. The multimodness introduces some ambiguity to the eigenvalue problem and make the optimization difficult to handle. Thus far, only the simplest cases of multimodal structures have been effectively optimized using rather elaborate analytical methods ([1-4]). Numerous publications report design of a minimum volume structure with different eigenvalues constraints, in which, however, the modality of the problem is assumed a priori (see [5-7], for example). The method presented here utilizes a multimodal optimality criteria and allows for inclusion of an arbitrary number of buckling or vibrations modes which might influence the optimization process. The real multimodality of the problem, that is the number of modes participating in the final optimal design is determined iteratively. Because of a natural use of the FEM technique the method is easy to program and might be helpful in design of large flexible space structures.

2 THE OPTIMALITY CRITERIA

The buckling and the free vibrations problems are formulated and solved using very similar numerical techniques. Consequently, the optimality criteria for those two cases must be also very similar. Here, due to space limitations, only the optimality criteria for the highest frequency of free vibrations is briefly outlined.

Using finite element formulation the free vibrations problem is defined by:

$$(K - \lambda_i M)x_i = 0 \tag{1}$$

where K is the elastic stiffness matrix, M the mass matrix, x_i is the i-th vibrations mode and λ_i represents the square of the corresponding frequency of free vibrations. Multiplying Eqn.(1) by x_i, the eigenvalue λ_i can be expressed in the form of the Rayleigh quotient as:

$$\lambda_i = \frac{1/2 \; x_i K x_i}{1/2 \; x_i M x_i} \qquad\qquad (2)$$

where the numerator represents the strain energy and the denominator represents the kinetic energy of the vibrating structure.

Using a proper eigenvalue solver the solution to the problem (1) is obtained in the form of the set of eigenvalues $\lambda_1, \lambda_2, \lambda_3 \ldots$, where $\lambda_1 \geqslant \lambda_2 \geqslant \lambda_3 \geqslant \ldots$, and the corresponding set of eigenmodes $x_1, x_2, x_3 \ldots$.

The purpose of optimization discussed here is to maximize the first eigenvalue λ_1 for a structure of prescribed weight. Clearly when increasing λ_1 it may become equal to λ_2 then to λ_3 and so on... Consequently the optimization finally may need to monitor N modes. Note that the number of modes which must be considered to reach the optimal design is unknown a priori and may be difficult to predict.

In order to derive an optimality criterion for the above problem consider the following Lagrange functional:

$$F(x,h) = (1/2 \; x_1 K x_1)/C_1 + \sum_{i=2}^{N} \gamma_i [(1/2 \; x_i K x_i)/C_i - (1/2 \; x_1 K x_1)]$$
$$+ \sum_{i=1}^{N} \eta_i \; (1/2 \; x_i M x_i - C_i) + \beta \; (\sum_j W_j - W_o) \qquad (3)$$

where h is the design variables vector representing a property (area, thickness, etc.) of each element to be optimized, C_i is an arbitrary constant to normalize the i-th vibrations mode, W_j and W_o are the weight of the j-th element and the total weight respectively, γ_i, η_i and β are the Lagrange multipliers.

The multipliers η_i and β can be determined explicitly and the necessary conditions for optimality of the functional (3), after some algebra, are derived in the form:

$$(1 - \sum_{i=2}^{N} \gamma_i) \; NSE^1_j + \sum_{i=2}^{N} \gamma_i \; [\frac{\lambda_i}{\lambda_1} (NSE^i_j - 1) + 1] = p-1 \qquad (4a)$$

$$\gamma_i (\lambda_i - \lambda_1) = 0 \qquad\quad i = 2 \ldots N \qquad (4b)$$

where: $\quad NSE^i_j = (W_o/W_j) \; (1/2 \; x_{ij} \hat{K}^i_j x_{ij}/\lambda_i C_i)$

and $\qquad \hat{K}^i_j = p K_j - \lambda_i M_j$

NSE^i_j is the normalized equivalent strain energy of the j-th element due to the i-th vibrations mode and x_{ij} must satisfy Eqn. (1). The parameter p represents the relation between the stiffness and the mass of each element which was assumed in the form:

$$(M_j)^p/K_j = const.$$

Eqn. (4a) must be satisfied for each element of the structure, and the switching conditions (4b) must be satisfied by every eigenvalue λ_i. As can be seen from Eqn. (4b) the number of nonzero Lagrange

multipliers, γ_i, indicates the modality of the problem. For the correct number of modes influencing the final optimal design the equations (4a) and (4b) can be combined into one condition:

$$(1 - \sum_{i=2}^{N} \gamma_i) \; NSE^1_j + \sum_{i=2}^{N} \gamma_i \; NSE^i_j = p-1 \tag{5}$$

The conditions similar to that given by Eqn. (5) were discussed in [5,6,7]. However, it should be emphasized that the modality of the optimal design represented here by N is, in general, unknown and therefore any iterative solution procedure must also simultaneously consider Eqn. (4b). Unfortunately, Eqn. (5) can be satisfied by a set of eigensolution which does not meet the conditions (4b). This features makes the optimization procedure based only on Eqn. (5) somewhat difficult to handle.

3 THE ITERATIVE PROCEDURE

The purpose of the iterative procedure is to meet the optimality conditions by correcting the design variables vector, h. The correction, δh_j to the shape of the j-th element is calculated using the following steps:

a) For a given structure solve the eigenvalue problem to obtain the solution with sufficient number of modes, N (we assumed N=9).
b) Utilizing the results from step (a) determine the energies NSE^i_j.
c) Select a "proper" set of the multipliers γ_i (see comments below)
d) Determine the local error (for the j-th element) defined as:

$$\xi_j = (1 - \sum_{i=2}^{N} \gamma_i) \; NSE^i_j + \sum_{i=2}^{N} \gamma_i \; NSE^i_j - p + 1 \tag{6}$$

e) Verify the optimality criteria against an assumed tolerance Δ (we used $\Delta = 0.0001$) e.g.:

$$|\xi_j| < \Delta \text{ for all elements and } \lambda_i/\lambda_1 - 1 < \Delta \text{ for } i = 2 \; .. \; N_R$$
$$\gamma_i < \Delta \text{ for } i = N_R+1 \; .. \; N \tag{7}$$

Since for $i > N_R$ all γ_i vanish, clearly, N_R represents the modality of the design.

f) If the optimality criteria are not met, correct the design variables accordingly to the formula

$$\delta h_j = c \xi_j h_j \tag{8}$$

where c is a positive number and repeat steps (a-e).

Note that in steps (a) and (b) only the magnitudes of NSE^i_j are calculated, while the Lagrange multipliers γ_i are still indeterminate. In fact, as long as $\lambda_1 < \lambda_i$ they may be chosen somewhat arbitrarily (see [10]). However, a unique set of γ_i must be determined when a design becomes the optimal one. The following two-phase procedure provides a good selection of γ_i in every iteration step and secures the convergence of the optimization process.

First the magnitudes of γ'_i are calculated minimizing the global error defined as $\sum \xi_j^2$. The minimization, when using Eqn. (6), leads to the set of linear algebraic equations in the form

$$\sum_{i=2}^{N} a_{ki}\, \gamma'_i = b_k \qquad\qquad k = 2 \ldots N \qquad\qquad (9)$$

where a_{ki} and b_k are known functions of NSE^k_j

The second phase implements corrections to γ'_i with respect to the difference between λ_i and λ_1 in the form:

$$\gamma_i = \gamma'_i\, (\lambda_1/\lambda_i)^{i-1} \qquad\qquad i = 2 \ldots N \qquad\qquad (10)$$

The relation (10) is used to satisfy iteratively the switching condition (4b) and plays a very important role in determination of the modality of the problem. For $i \leqslant N_R$ the difference between γ_i and γ'_i gradually disappears while for $i > N_R$ the values of γ_i are consistently driven to zero the same eliminating the corresponding modes from optimization.

4 SOME RESULTS AND DISCUSSION

The procedure discussed here has been used for optimal design of columns, frames and plates. Some unimodal and bimodal cases of the optimization for maximum stability were presented in [8-10]. The optimization program interacts with ANSYS, the FEM software, which performs the analysis required in step (a).

Fig. 1 shows the examples of bimodal and trimodal designs for maximum stability. Fig. 2 presents the case of optimal design of the plate for maximum frequency. In both cases the optimization was initiated from the uniform shape and the increase in the buckling load or in the frequency of free vibrations is also related to the corresponding parameter for the uniform structure.

The numerical experimenting indicates that the method is globally convergent, though the process requires a high accuracy in calculations of eigenmodes in every iteration.

REFERENCES

[1] N. Olhoff and S.H. Ramussen. On single and bimodal optimum buckling loads of clamped columns. Int. J. Solids Struct., 11, 605-614 (1977).
[2] J. Blachut and A. Gajewski. On unimodal and bimodal optimal design of funicular arches. Int. J. Solids Struct., 17, 653-667 (1981).
[3] R.H. Plaut, L.W. Johnson and N. Olhoff. Bimodal optimization of compressed columns on elastic foundations. J. Appl. Mech. 53, 130-134 (1986).
[4] B. Bochenek. Multimodal optimal design of a compressed column with respect to buckling in two planes. Int. J. Solids Struct., 23, 599-606 (1987).
[5] R.V. Grandhi and V.B. Venkayya. Structural optimization with frequency constraints. AJAA Journal, 858-866 (1988).

[6] N.S. Khot, V.B. Venkayya, L. Berke. Optimum structural design with stability constraints. Int. J. Numer. Math. in Engrg., 10, 1097-1114 (1976).

[7] N.S. Khot, Optimization of structures with multiple frequency constraints. Computers & Structures, 20, 868-876 (1985).

[8] W. Szyszkowski, L.G. Watson. Optimization of the buckling load of columns and frames. Engrg. Struct., 10, 249-256 (1988).

[9] W. Szyszkowski, L.G. Watson, B. Fietkiewicz. On the multimodal optimization of plates against buckling. ASME PVP-149, 1-7 (1988).

[10] W. Szyszkowski, L.G. Watson, B. Fietkiewicz. Bimodal optimization of frames for maximum stability. Computers & Structures, 32, 1093-1104 (1989).
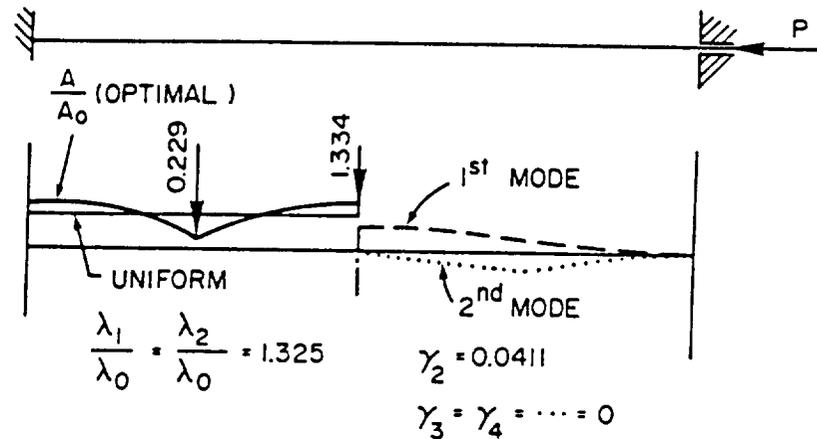
(a) BIMODAL OPTIMIZATION OF FIXED - FIXED COLUMN
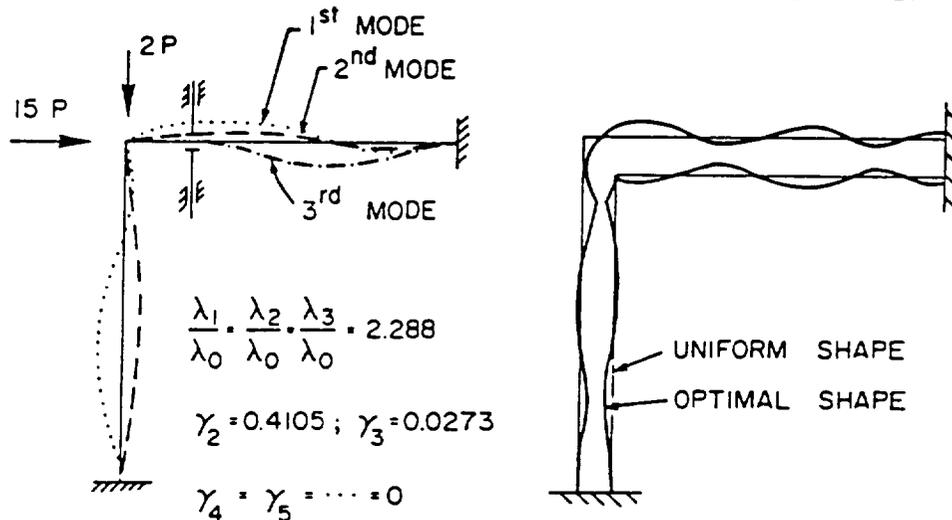
(b) TRIMODAL OPTIMIZATION OF FRAME

FIG. I EXAMPLES OF MULTIMODAL OPTIMIZATION FOR MAXIMUM STABILITY

301

(a) SQUARE PLATE WITH NONSTRUCTURAL
   MASS AT THE CENTER

(b) OPTIMAL SHAPE
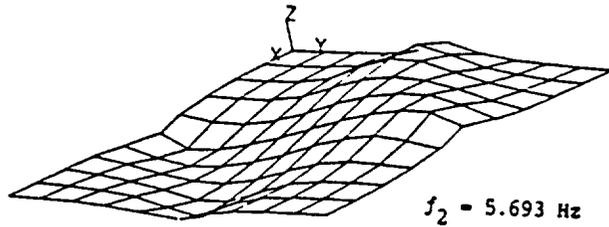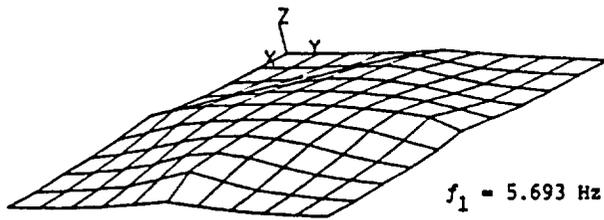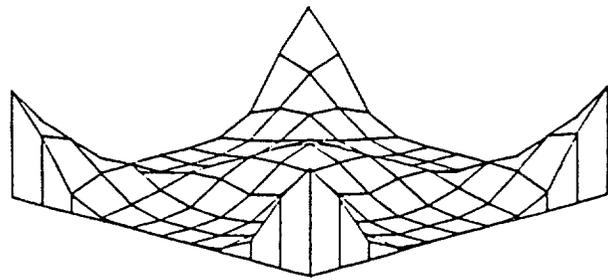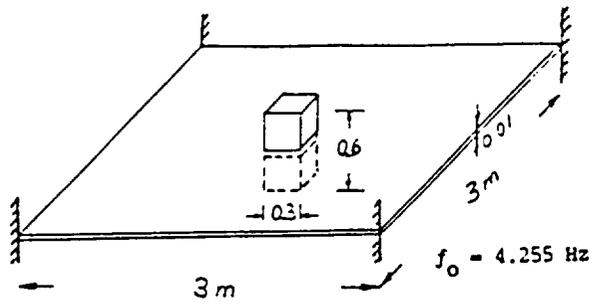
(c) MODES OF VIBRATIONS
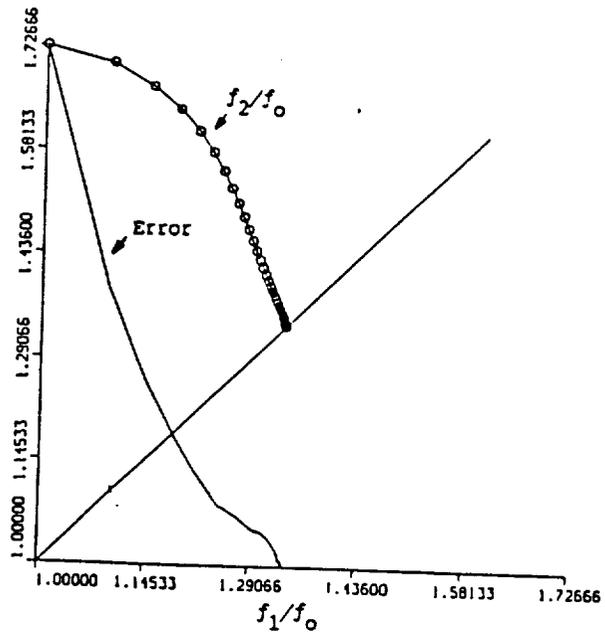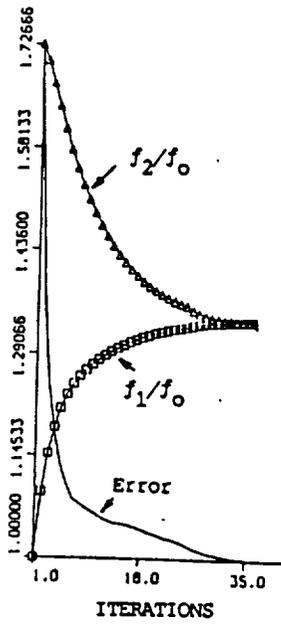
(d) CONVERGENCE HISTORY

FIG. 2 BIMODAL OPTIMIZATION FOR VIBRATIONS

THERMAL STRESS ANALYSIS OF THE NASA DRYDEN HYPERSONIC WING TEST STRUCTURE

GLENN MORRIS, ENGINEERING SPECIALIST

GENERAL DYNAMICS CORPORATION, FORT WORTH DIVISION
P.O. BOX 748, MAIL ZONE 2829
FORT WORTH, TEXAS 76101

KEYWORDS: Finite Element Method, Finite Difference Method, Thermal Stress Analysis, Hypersonic Wing Test Structure, Hypersonic Aircraft.

ABSTRACT

Present interest in hypersonic vehicles has resulted in a renewed interest in thermal stress analysis of airframe structures. While there are numerous texts and papers on thermal stress analysis, pracical examples and experience on light gage aircraft structures are fairly limited. A research program has been undertaken at General Dynamics to demonstrate the present state of the art, verify methods of analysis, gain experience in their use, and develop engineering judgement in thermal stress analysis. The approach for this project has been to conduct a series of analyses of this sample problem and compare analysis results with test data. This comparison will give an idea of how to use our present methods of thermal stress analysis, and how accurate we can expect them to be.

INTRODUCTION

Several strategies for thermal stress analysis have been used in the past. Prior to the emergence of computerized analysis, numerous texts were written on closed form (or "classical") methods (see Reference 1). When these have been used, such as on the X-15, X-20, and B-58, several drawbacks occurred: 1)Length on time required to perform calculation made it difficult to converge on a design (Reference 2), 2)Unknowns such as joint flexiblity and simplifying assumptions often led to overprediction of stresses (Reference 3) or surprising failures (Reference 4).

With the emergence of digital computers, the finite element method came into prominence for structural analysis. For thermal analysis, the finite difference method has become the primary computer method. This resulted in more detailed analysis in both specialites. However, during the late 1960's and the 1970's, there was little activity in thermally stressed airframe structures. For example, the original design philosophy of the space shuttle was that

the TPS (Thermal Protection System) would keep thermal stresses from devoloping in the orbiter structure. This assumption has proven to be false (References 5 through 7). Reference 5 briefly discusses some of the problems of thermal stress analysis as done on the Space Shuttle Orbiter. Thermal Analysis was done on 118 local lumped parameter (finite difference) models. Most temperatures used in stress analysis were obtained by interpolating between these models. The potential for error at this point in the analysis is high, especially if thermal analysis models are not in the right place or if the distance between them is too great. NASA has concluded that more detailed thermal models are needed. This in turn requires: 1) automating methods of thermal model generation (similar to finite element preprocessors), 2) faster solution techniques, 3) automating the search for critical conditions, 4) improved modeling techniques to reduce model size, 5) automating data transfer and/or interpolation between thermal analysis and structural analysis (similar to finite element post processors), and 6) more reliable and accurate cacaulation of aerodynamic heating inputs.

To achieve these goals, NASA Langley Research Center has begun developing a fully integrated finite element method, encompassing structural, thermal, and aerodynamic analysis into a single solution. The main reasons for this strategy are 1) to solve the problem of interfacing the different discipines by eliminating the interface, and 2) taking advantage of the model generating capablities of existing finite element pre/post processors. While this work is still in the development stage, it shows great promise (see References 8 through 10). The method has tentatively been named LIFTS (Langley Integrated Fluid-Thermal-Structural Analysis).

Until LIFTS or other new methods are fully operational, we must work with existing methods. This means using finite element methods such as NASTRAN for structural analysis, finite difference methods (such as SINDA) for thermal analysis, and either approximation methods (such as AEROHEAT) or CFD (computational fluid dynamics) methods for aerodynamic heating analysis. The problem then becomes one of

interfacing and coordinating the different analyses. Our first attempts quickly showed that if each task was done independently, without coordination and cooperation, results would be dismal. It was therefore decided that all three analyses must be worked on a common grid layout, and that the interface between them would be automated by a computer program to generate a SINDA model from a NASTRAN model, and to reformat SINDA output for use by NASTRAN. This computer program, developed by General Dynamics under discretionary funding, is called NASSINDA.

Once this procedure was operational, it was decided to exercise it on a test problem. This was intended to demonstrate our capablities, to develop experience and technique in this type of analysis, and to find bugs in the code. The test problem needed to be fairly representative of airframe structure, well documented, and well instrumented.

HWTS BACKGROUND

The problem chosen for analysis was the Hypersonic Wing Test Structure (HWTS). HWTS is a metallic structure simulating the wing of a hypersonic aircraft (see Figure 1). The design of the HWTS was based on the mission loads and temperatures calculated for the wing portion of the Hypersonic Research Airplane (HRA), a study vehicle which was never built.

A major design consideration was the pushover - pullup loads maneuver. This maneuver is initiated at Mach 8 at an altitude of 27.4 km, and consists of a - 0.5 G pushover, a +2.5 G pullup, and a return to a nominal research mission descent profile. Preload and postload maneuvers precede and follow the actual loads maneuver, providing transitions to and from the nominal flight path. The entire loads maneuver took 42 seconds, with a maximum dynamic pressure of 83.78 kn/m**2 obtained during this time.

TEST ARTICLE

The following description of the HWTS test article is based on Reference 11 and engineering drawings of the HWTS.

Figure 2 shows the general dimensions and shape of the HWTS with a transition section. The wing is cantilevered from W.S. (wing station) 42. 00 (1 . 067M). The wing was tested inverted, so the compressively loaded surface of the actual vehicle would be the lower surface of the test structure. Although not a part of the airplane design, the transition section was included to provide a buffer between the support structure and the test portion of the wing. The five most critically compression-loaded panels are the lower root panels.

Skin panels of the HWTS are the primary load carrying members. They are formed from a single sheet of Rene' 41 with seven alternating up and down circular arc beads parallel to the wing spars. Doublers were spot-welded to the ends of the panel to prevent local end failure and to reduce excessive deformation caused by shear. Overall panel dimensions are 19.25 inches (48.8 cm) by 42.9 inches (109.0 cm). Beads are .026 inches (.066 cm) thick having a radius of 1.045 inches (2.654 cm) with an included angle of 155 degrees. The flat sections between the beads are .438 inches (1.113 cm) wide and .036 inches (0.081 cm) thick. Four channel sections are spot welded to the beaded panel to provide attachment points for metallic heat shields. The beaded panels are attached to the caps of orthogonal spars and ribs by screws.

Figure 2 shows the HWTS mounted in a support fixture. Z-shaped clips are used to connect the heat shields to the structure. The HWTS has. six spars perpendicular to the aircraft centerline, producing five chordwise bays. Both the spar and rib webs have sine wave corrugations allowing for thermal expansion. The outboard portion of the structure between the leading edge rib and the 30% rib is covered by an insulation packet; the insulation is intended to keep maximum structural temperatures below 1005 degrees K and to keep spanwise temperature gradients constant.

The heat shields are slightly corrugated in the chordwise direction. In general, two heat shields cover each full - size beaded panel. Heat shield extensions were provided around the boundaries of the test structure to improve the heating simulation of the outer spars and rib webs.

Figure 2 shows the attachment of the wing to the support structure. Twelve horizontal links provide spanwise horizontal load reaction and reaction for bending moment about an axis parallel to the aircraft centerline. Each link has a spherical ball at each end so that thermal growth of the wing is not restricted by the support structure. Water-cooled fittings were placed between the links and the wing to maintain the support structure at room temperature during tests. There is also an insulation packet that extends along the wing root to prevent heat losses.

Three sets of testing have been done on the HWTS. Originally, it was believed that lateral pressure on the beaded skin panels was an important consideration. A "pressure pan" was placed under each of the critical lower surface root panels, and pressure was applied to create a lateral load on the panel. This is the series of tests described in Reference 12. Later, these tests were repeated with a set of panels of alternate design. Finally, attention shifted away from the panels and to the global thermal stress analysis problem. The pressure pans

block radiation coming off the backside of the skin panels, so they were deleted from the third, or "Retest" series of tests. This data has not been formally reported by NASA, but was made available for our use (Reference 13).

TEST INSTRUMENTATION

A total of 370 thermocouples, 102 strain gages, and 18 deflection pots were installed on the HWTS for the "Retest" series of tests. Some 78 of the strain gages were conventional foil gages, which are limited to temperatures of 550 degrees or less. The remaining 24 strain gages are weldable strain gages, which are operational at higher temperatures. This has an impact on the test conditions which will be discussed later.

Instrumentation is concentrated in and around Bay "H" (Figure 3), which is bounded by spars at F.S. 950 (24.130M) and F.S. 970 (24.638M) and ribs at B.L. 54 (1.372M) and B.L. 97.835 (2.479M). This bay is one of the inboard bays, and therefore heavily loaded (The wing root is at B.L. 54.0). It is also as far as possible from the thermal boundaries, making the thermal simulation in this area as realistic as possible. Thus, this area is the center of attention.

TEST PROCEDURE

The "retest" series of tests consisted of three test runs. First was a check-out run to verify operation of instrumentation, loading system, heating system, and controls. Only room temperature deflection data is available from this run (Condition 4.1.3). The second test was a combination of thermal and mechanical loads, with the temperatures limited to 550 degrees F (Condition 550.4). This insured that temperature limits of strain gages were not exceeded. The third test was a combination of mechanical loads and thermal load, similar to the second test, except that temperatures were limited to 1100 degrees F (condition 1000.4). This means that the foil strain gages were destroyed sometime during the test, with only the 24 weldable strain gages operational throughout the test.

THERMAL STRESS ANALYSIS VERIFICATION

The object of the HWTS investigation at General Dynamics was to exercise our methods of thermal stress analysis, debug them, and develop experience using them in realistic airframe structure applications. Our approach has been to run a series of demonstration problems based on data from the HWTS tests just described. Comparision of test and analytical results helps give an understanding of the accuracy of our calculations.

CHECK PROBLEM ONE - BAY H OF HWTS

Bay H, one of bays on the HWTS, was chosen as the basis of our first checkout problem for a variety of reasons. It is one of the most inboard bays and therefore most heavily loaded mechanically. It is the bay farthest

from the boundaries and therefore most realistic in its thermal simulation. It is the most heavily instrumented area of the HWTS. It is also a small enough problem to serve as a good checkout before starting on analysis of the full HWTS. In Reference 14, Lameris used Bay H to check out problems in the COSMIC NASTRAN thermal analyzer. A copy of that NASTRAN deck was provided by NASA Dryden. Our analysis was made using the same grid layout.

NASSINDA does not recognize the NASTRAN heat transfer (CHBDY) elements used by Lameris, so they were replaced by dummy CQUAD4 and CTRIA3 elements which represent the heat shield. The model is illustrated in Figure 4.

The first run of the SINDA model was incorrect due to an input data error. The second run produced skin temperatures in good agreement with measured temperatures in Reference 14, but understructure (spar webs) temperatures were too high. Since the spar webs are corrugated in a sine wave pattern, they have a higher conductance in the vertical direction than the flat plate values used in this run. Also, an emissivity value of .8 was used, which was judged to be too high.

For the third run, the conductances of the spar webs was increased in the vertical direction. This resulted in an increase in cap temperatures and a reduction of web tempertures. However, the lower web temperatures were still too high early in the trajectory. For the fourth run, emissivity was changed from .8 to .65. Good agreement was achieved on understructure and skin temperatures, as shown in Figure 5. This gave us enough confidence to procede with an analysis of the full HWTS.

CHECK PROBLEM TWO - FULL HWTS

Since the thermal analysis model of Bay H was successful, the next test problem was a full three dimensional model of the HWTS, shown in Figure 6. In its layout, it is similar to a typical coarse grid internal loads finite element model. Load carrying skins are modeled as CQUAD4 elements, spar and rib vertical webs are represnted by CSHEAR elements, and CROD elements model spar and rib caps. The model was checked out by comparing its deflections under static load with those from test.

The General Dynamics MODGEN program generated the basic connections of this model. Additional work done by text editing included:

Definition of element thicknesses and areas - Data read from drawings was used.

Definition of radiation enclosures - Radiation view factor calculations need information on which parts of the structure "see" other parts. This data is included in NASSINDA input data. CQUAD4 elements 740 thru 780 were added

to complete the enclosure between the skin and the heat shield.

Definition of Boundary Conditions - The HWTS is mounted on a test fixture made of structural steel. This was added to the model, and tied to the wing box with NASTRAN MPC's.

Leading Edge Insulation - The insulation between the leading edge heat shield and skin was modeled using CHEXA elements 780 thru 794.

Mechanical Load - Force cards have been added to simulate test load conditions. Force set 41 is condition 550.4 (retest).

Thermal Load - After using NASSINDA to generate the SINDA model, thermal loads were added by imposing a temperature on the heat shield panels in the SINDA data deck.

The first SINDA run had an error in its input temperature load (a missing decimal point) causing one input to be read as zero and rendering the run incorrect. The second SINDA run showed unusual temperature contours around the leading edge. The CHEXA elements used to represent insulation were 1.5 inches thick, but the real insulation blanket was only 1/8 inch thick. Properties of the insulation material were adjusted to account for this.

SINDA calculated temperatures from the third run show good agreement with test data in most areas. Several channels of thermocouple data appear to be in error. For example, the lower skin panel in the outboard forward corner shows a test temperature of 239.7 degrees F. This area is insulated, and should be closer to 101 degrees F. Similarly, a panel of the upper skin showing an 83.5 degrees F test temperature in an uninsulated area should be closer to 240 degrees F. For comparision purposes, an "assumed true" temperature was used in locations where test data was apparently incorrect, or was not taken.

For most of the wing box, calculated skin temperatures are 10 to 15 degrees F higher than the test temperatures. The errors for spar and rib cap calculated temperatures are 50 to 140 degrees F too low. These errors give large fictitious thermal stresses. The cause of this error is the large mesh size used. Heat radiated from the heatshield to the skin goes to a thermal node in the middle of that skin bay. Heat can then be conducted to the spar caps, or radiated to the understructure webs. Since radiation is, in fact distributed over the panel, the conduction path between the panel and the cap is too long. Also, since the spar cap blocks backside radiation for 3 inches (15%) of the 20 inch wide bay, these modeling errors combine to make radiation heat transfer from the skins to webs too high and conduction heat transfer from the skins to caps too low.

There is also a radiation path to and from the caps which is ignored. Since the understructure webs are formed as sinewaves, their true mass is greater than the flat panel mass in the model.

Calculated temperatures in the "transition bay" (BL 42.0 to BL 54.0) are higher than measured temperatures. This may be because the root attachment fittings were not modeled. These fittings are a sizeable mass of stainless steel, and they were water cooled.

As a first attempt to correct this, emissivity was decreased from .65 to .63 on the next (fourth) run. Also, mass of the understructure webs was corrected and the water cooled root fittings were simulated by holding the nodes they connect to at a constant 80 degrees F. A comparision of calculated and measured skin temperatures is shown in Figure 7 and temperatures for a spar are shown in Figure 8. These understructure temperatures are still not as accurate as desired.

To test the impact of internal radiation on the cap temperatures, the grid was modified in two bays as shown in figure 9. The small strips were added to the edges of two panels of the upper skin. These panels receive radiation from the heat shield. They do not radiate on the back side, since that area is covered by the caps. Therefore, that heat is conducted to the caps. The temperature on the cap under the strips did rise about 40 degrees F. This indicates that this is a source of error in the cap temperatures.

Because of the impact of mesh refinement on the cap temperatures, two new grids were laid out, as shown in Figure 12. This work should give us enough data for a mesh convergence study, and tell us what general level of refinement is needed to get meaningful results.

CONCLUSIONS TO DATE

First, thermal analysis and structural analysis must be performed in close coordination and cooperation. The analysis grid must be chosen with consideration for both heat flux and load paths. Data transfer between the two disciplines must be automated to solve problems of practical airframe size.

Secondly, radiation heat transfer is a tricky, time consuming part of the problem. Radiation view factor calculation is a major part of the problem.

Thermophysical material properties, such as emissivity, conductance, and capacitance, are not as easily available as mechanical properties. Often these are temperature dependent, making an exact solution nonlinear.

Finally, several studies of structure this size and numerous smaller studies are needed before we can reasonably expect to release hot airframe structure for hypersonic flight.

REFERENCES

1. "Thermal Stresses", B. E. Gatewood, McGraw-Hill Book Company, New York, 1957.

2. ASIAC Report No. 685.1a, "Lessons Learned in the Structures, Dynamics, and Materials Area during the X-20 (Dyna-Soar) Program", Richard B. Baird, Aerospace Structures Information and Analysis Center, AFWAL/FIBR, Wright-Pattrson AFB, Ohio 45433, September 1986.

3. Personal Conversations with Dale Marshall and Larry Seth of the B-58 stress analysis team.

4. "Structural Design of the X-15", Journal of the Royal Aeronautical Society, pages 618 through 636, R. L. Schleicher, October 1963.

5. NASA TM-86420, "NASA R&D for Aerospace Plane Vehicles - Progress and Plans", S. C. Dixon, May 1985, page 74.

6. NASA TM-88292, "Effect of Element Size on the Solution Accuracies of Finite-Element Heat Transfer and Thermal Stress Analyses of the Space Shuttle Orbiter", W. L. Ko and T. Olona, August, 1987.

7. NASA CP-2216, "Computational Aspects of Heat Transfer in Structures", H. M. Adelman, Compiler, 1982.

8. NASA TM-86434, "A New Finite Element Approach for Prediction of Aerothermal Loads - Progress in Inviscid Flow Computations", Kim S. Bey, Earl A. Thornton, Pramote Dechaumphi, R. Ranakrishnan.

9. "Coupled Flow, Thermal, and Structural Analysis of Aerodynamically Heated Panels", E. A. Thornton and P. Dechaumphai, AIAA/ASME/AHS 28th Structures, Structural Dynamics and Materials Conference, Monterey, California, April 6-8, 1987, AIAA Paper No. 87-0701.

10. "Flow-Thermal-Structural Study of Aerodynamically Heated Leading Edges", P. Dechaumphai, E. A. Thornton, A. R. Wieting, AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics, and Materials Conference, Williamsburg, Virginia, April 18-20, 1988, AIAA Paper No. 88-2245-CP.

11. NASA TM-85918, "Comparision of Measured and Calculated Temperatures for a Mach 8 Hypersonic Wing Test Structure", R. D. Quinn, R. A. Fields, NASA Dryden Flight Research Facility, Edwards, California, March 1988.

12. NASA CR-127490, "Hypersonic Wing Test Structure Design, Analysis, and Fabrication", P. P. Plank, and F. A. Penning, Martin Marietta Corp, Denver, Colorado, August, 1973.

13. "Retest" Data Package recieved from NASA Dryden, February 1987.

14. NASA CR-170413, "Development of a Thermal and Structural Model for a NASTRAN Finite-Element Analysis of a Hypersonic Wing Test Structure", Jaap Lameris, University of Kansas Center for Research, Lawrence, Kansas, February, 1984.
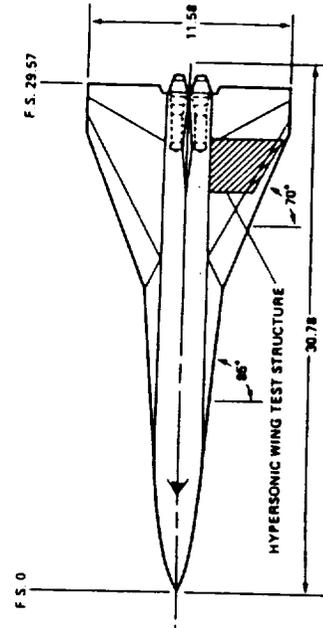
FIGURE 1 - HYPERSONIC RESEARCH AIRCRAFT
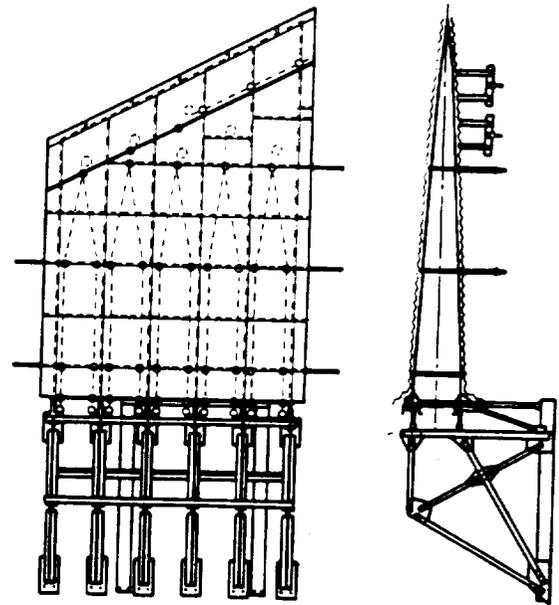
FIGURE 2 - HYPERSONIC WING TEST STRUCTURE (HWTS)

FIGURE 3 - HWTS THERMOCOUPLES ON BAY H UPPER SKIN

FIGURE 5 - HWTS BAY H TEMPERATURES

FIGURE 4 - HWTS BAY H ANALYSIS GRID

FIGURE 8 - ANALYSIS GRID OF HWTS

HWTS SKIN TEMPERATURES -- MEASURED VERSUS CALCULATED

UPPER SKIN

119.4 (105.0)	115.1 (109.3) [240.0] 211.1	111.0 (114.8) 83.5 (305.2)	125.5 (120.3) 333.0 (349.4)	136.9 (126.2) 396.7 (399.3)
244.9 (229.5)	289.0 (298.9)	281.5 (298.9)	311.5 (339.8)	380.4 (401.9)
253.6 (226.6)	261.5 (256.1)	311.5 (294.3)	314.2 (336.7)	366.1 (385.8)
168.2 (212.5)	197.9 (240.0)	223.1 (275.5)	250.1 (314.8)	264.1 (363.0)

LOWER SKIN

239.7 [101.0] (112.6)	101.1 (117.0) [240.0] 346.0	102.0 (122.8) 267.2 (381.3)	109.8 (128.7) 439.8 (422.9)	126.5 (134.9) 455.1 (471.6)
253.6 (291.4)	[340.0] (522.6)	342.0 (367.6)	417.1 (412.1)	438.1 (479.5)
258.9 (286.2)	286.3 (319.8)	334.8 (362.8)	383.3 (409.1)	424.7 (462.2)
239.7 (292.7)	225.7 (325.0)	291.7 (368.0)	307.9 (414.6)	381.7 (470.4)

XXX - TEST TEMPERATURE   (XXX) - SINDA CALCULATED TEMPERATURE   [XXX] - ASSUMED TRUE VALUE
TEMPERATURES FOR CONDITION 550.4 (RETEST) AT 400 SECONDS
EMISSIVITY = .63, WEB MASS CORRECTED FOR SINE WAVE SHAPE

FIGURE 7 - SKIN TEMPERATURES - CALCULATED VS. MEASURED

COARSE GRID MODEL GIVES LESS ACCURACY IN UNDERSTRUCTURE

HWTS SPAR 950

143.8 (87.9)   222.3 (90.3)   228.1 (90.9)   (90.9)   92.5 (81.6)

(115.7)   138.6 (158.8)   142.1 (174.7)   (151.1)   83.5 (93.9)

(92.9)   (94.1)   (94.6)   (96.9)   (82.0)
152.5   176.8   197.9   311.5 [90.0]

XXX - TEST TEMPERATURE
(XXX) - SINDA CALCULATED TEMPERATURE
[XXX] - ASSUMED TRUE TEMPERATURE

EMISSIVITY = .63, WEB MASS ADJUSTED FOR SINE WAVE SHAPE

FIGURE 8 - SPAR TEMPERATURES - CALCULATED VS. MEASURED

309
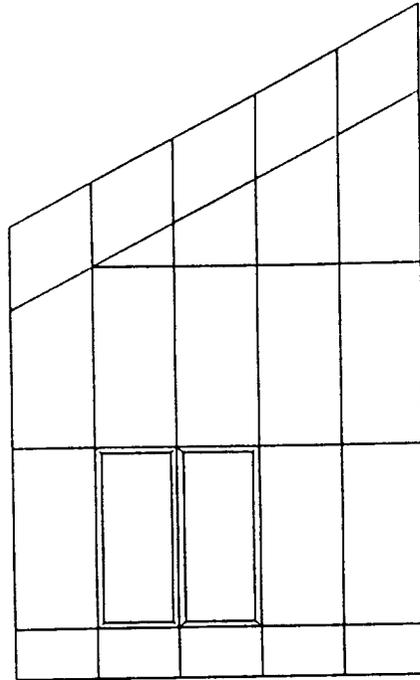
FIGURE 9 - LOCAL REFINEMENT OF ANALYSIS GRID
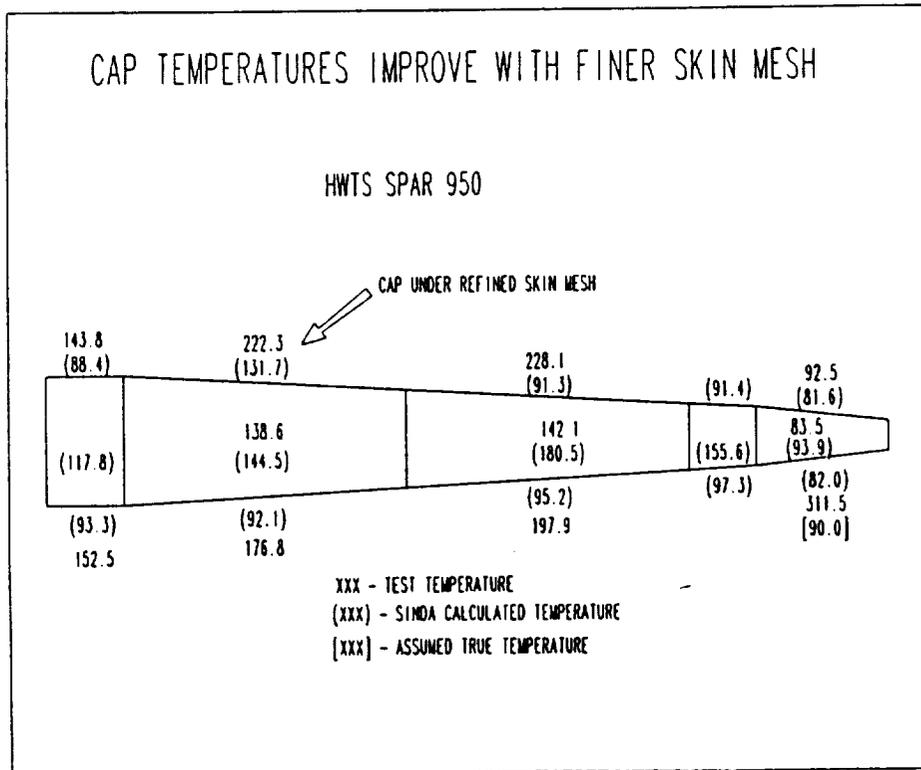
FIGURE 10 - SPAR TEMPERATURES AFTER LOCAL REFINEMENT

310
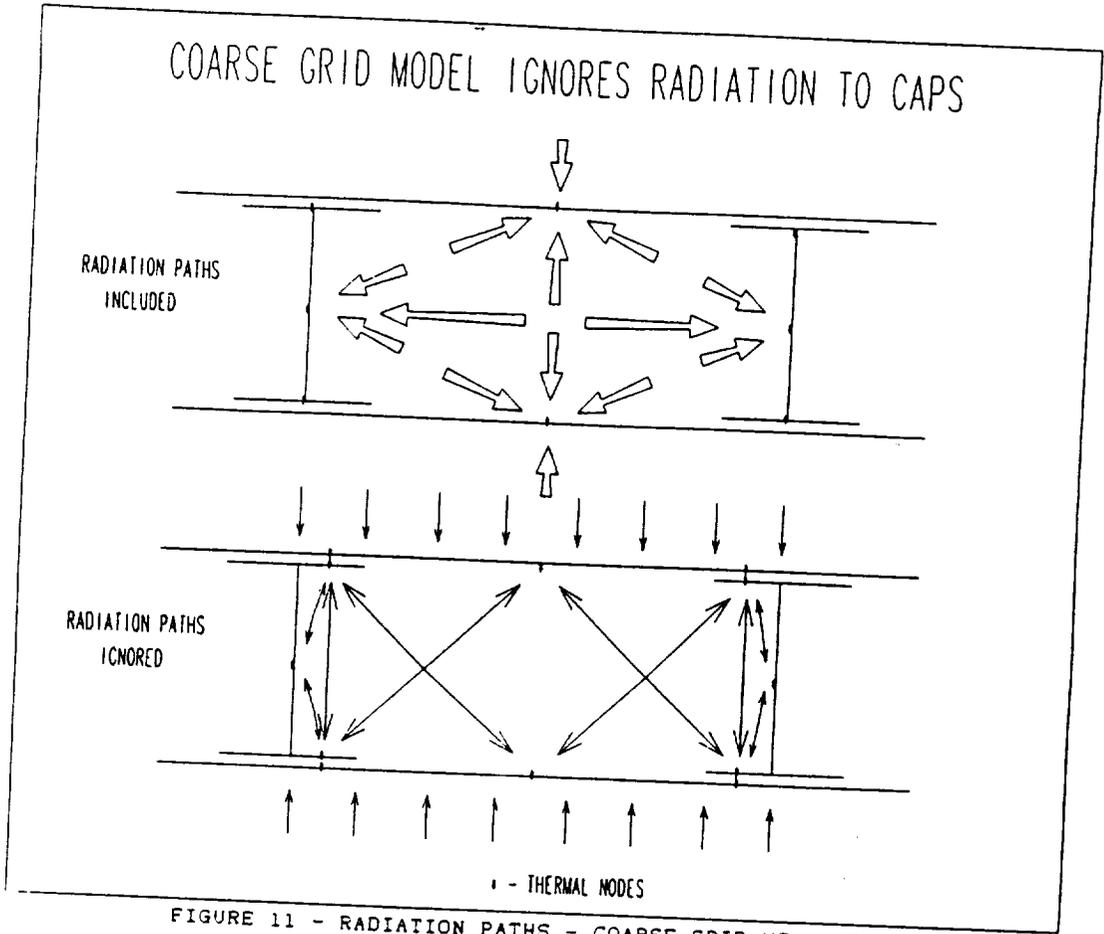
FIGURE 11 - RADIATION PATHS - COARSE GRID VS. ACTUAL
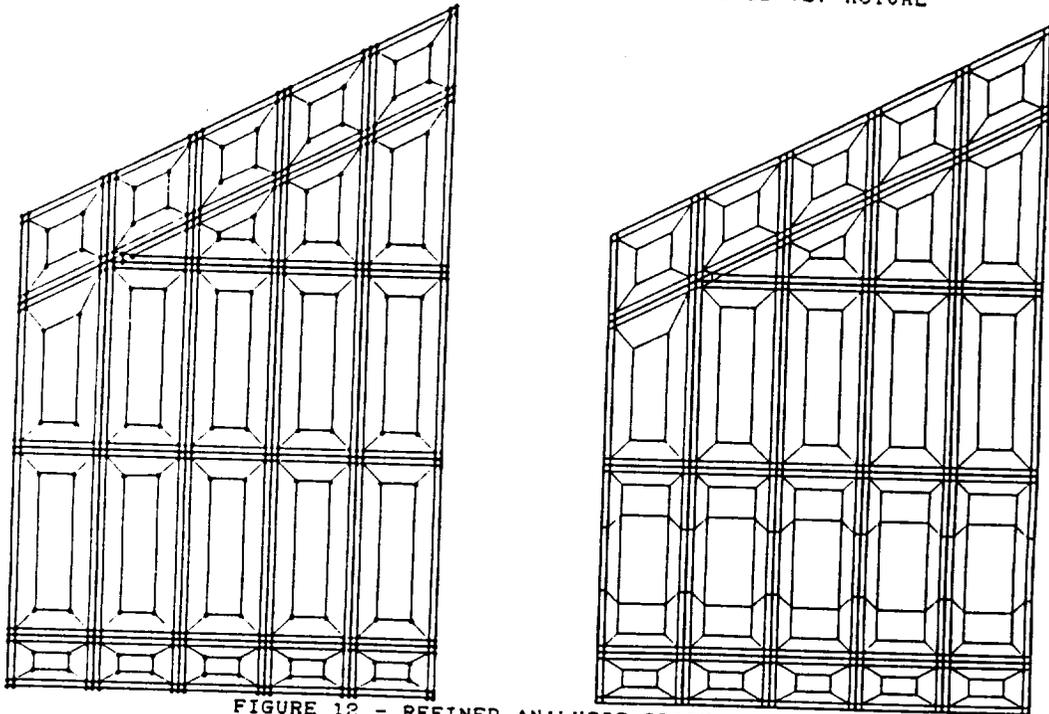
FIGURE 12 - REFINED ANALYSIS GRIDS OF HWTS

311

WEIGHT MINIMIZATION OF A SUPPORT STRUCTURE

Donald J. Kluberdanz
Helaine J. Segalman

Naval Underwater Systems Center
New London Laboratory
New London, Connecticut 06320

ABSTRACT

This paper addresses the weight minimization of a circular plate-like structure which resulted in a 26% weight reduction. The optimization was performed numerically with the COPES/ADS program using the modified method of feasible directions. Design parameters were the inner thickness and outer thickness of the plate with constraints on maximum yield stress and maximum transverse displacement. Also, constraints were specified for the upper and lower bounds of the fundamental frequency and plate thicknesses. The MSC/NASTRAN finite element program was used for the evaluation of response variables. Original and final designs of the plate were tested using an Instron tension-compression machine to compare finite element results to measured strain data. The difference between finite element strain components and measured strain data was within engineering accuracy.

INTRODUCTION

Weight minimization of a 13.6 inch diameter aluminum plate-like structure with 37 holes was performed resulting in a 26% weight reduction. The original cross section had a uniform thickness of approximately 2 inches. Boundary conditions consisted of the lower outer edge being simply supported. The structure was subjected to a uniform pressure over a portion of the surface totaling 23.0 kips (see Figure 1).

The purpose of this study was to numerically determine the minimum weight structure which would satisfy constraints on the following performance characteristics; fundamental frequency, maximum yield stress, and maximum transverse displacement. This paper demonstrates the application of numerical optimization using the COPES/ADS optimization program in conjunction with the MSC/NASTRAN finite element program.

PROBLEM FORMULATION

The objective was to minimize the weight of the structure. Overall dimensions, loading, and boundary conditions are described in the introduction and are shown in Figure 1. The design variables were the thickness at the center and thickness at the outer perimeter of the plate with a linear variation in the radial direction. The thicknesses

were bounded from 0.1 to 10.0 inches to ensure that a physically meaningful solution be obtained (i.e. the thickness at any point does not take on a negative value). Constraints imposed on the design were as follows; maximum Von Mises stress be below the 40.0 ksi yield stress of aluminum, maximum transverse displacement be below 0.117 inches, and the fundamental frequency be within 15% of a specified valve. All constraints were normalized during the optimization for better numerical conditioning.

The weight minimization was performed with the COPES/ADS optimization program using the modified method of feasible directions. Approximations were created for the objective and response variables using Taylor series expansions. At least three designs were required to generate a first order approximation. The original design was evaluated and two other designs were obtained by perturbing the design variables and analyzing them. This technique is not efficient for problems with a large number of design variables, but works well here for this two design variable problem.

FINITE ELEMENT MODEL AND TEST MODEL

The MSC/NASTRAN finite element program was used to evaluate the response variables during optimization. The structure was discretized into 532 twenty noded continuum elements, CHEXA elements in MSC/NASTRAN. It wasn't possible to model the structure with plate elements because it did not meet a ten to one length to thickness ratio typically used for a thick shell. Due to existing symmetry in the geometry, boundary conditions, and loading, only one-eighth of the structure was modeled with the appropriate symmetry boundary conditions. The finite element model was sufficiently refined to describe performance characteristics needed for optimization. It was necessary to further refine the model at the strain gage locations for a more detailed representation of the strain at these points.

The original and final designs of the structure were tested using an Instron tension-compression machine to compare finite element results to measured strain data. Both the uniformly thick and tapered structures were placed on a support ring to simulate the simply supported boundary conditions. A thick piece of rubber was placed between the structure and the load plate. The Instron load cell applied a force to the load plate and strain data was recorded at gage locations.

DISCUSSION OF RESULTS

Optimization of the plate-like structure reduced the weight from 13.5 lbs to 10.0 lbs, resulting in a 26% reduction. Both the initial design and final tapered configuration satisfy all the governing constraints or was said to be feasible. Figure 2 lists the geometry and constraint information for all six designs (t1 is the plate thickness at the center and t2 is the outer thickness). Note that once the optimization was initiated (designs 4-6) each iteration has some weight reduction as well as satisfying the feasibility criteria.

313

For the original configuration the maximum Von Mises stress of 33.9 ksi occurs around the center hole on the top and bottom surfaces. For the final design the maximum yield stress of 37.6 ksi has been shifted outwards due to the taper. The maximum transverse displacement for the original and optimized designs are 0.025 inches and 0.034 inches, respectively. The maximum transverse displacement for each design occurs in the center and the displacement field is radially symmetric. The fundamental frequency of the optimized model was within 15.3% of the required value. Although this design constraint states that the fundamental frequency be within 15.0% this was close enough to be considered acceptable.

A comparison of finite element strain values and experimental strain values for the original design and final design are shown in Figure 3. The strain components are for a total load of 23.0 kips, however the load was applied incrementally to the plates. The test results were linear and repeatable for a second cycle. The difference between finite element strain components and measured strain data was within engineering accuracy.

STRUCTURAL OPTIMIZATION
OF THE PLATE-LIKE STRUCTURE

FINITE ELEMENT MODEL
OF EXISTING STRUCTURE
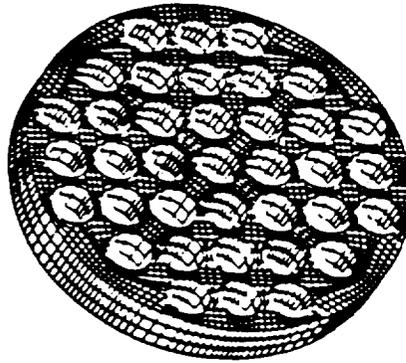
OPTIMIZED CROSS-SECTION
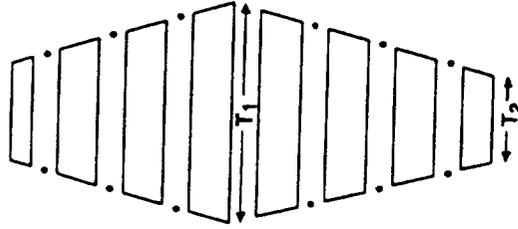
- UNIFORM SURFACE PRESSURE OVER A RADIUS OF 3.8"
  FOR A TOTAL LOAD OF 23,000 LBS

- OPTIMIZED PLATE WAS REDUCED IN WEIGHT BY 26%

FIGURE 1

SUMMARY OF DESIGNS

Design	t1 (in)	t2 (in)	Max Von Mises (ksi)	Freq. *Hz/Hz%	Max Disp. (in)	Weight (lbs)
1	2.065	2.065	33.9	1.1	0.025	13.5
2	2.200	1.800	33.2	4.0	0.027	12.5
3	1.700	2.000	41.6	7.9	0.033	12.6
4	2.105	1.600	37.9	10.6	0.033	11.4
5	2.229	1.400	37.6	13.5	0.035	10.7
6	2.529	1.130	37.6	15.3	0.034	10.0

* % of Required Value

FIGURE 2

COMPARISON OF RESULTS

ORIGINAL DESIGN

Gage	Test Strain μ in/in	FEM Strain μ in/in	Difference %
1	1928	2147	11
2	1754	1873	7
3	1180	1390	18
4	478	462	3
5	519	541	4

OPTIMIZED DESIGN

Gage	Test Strain μ in/in	FEM Strain μ in/in	Difference %
1	2610	1814	30
2	2660	2050	23
3	2299	1689	27
4	536	431	20
5	636	570	10

FIGURE 3

Genetic-Evolution-Based Optimization Methods for Engineering Design

S.S. Rao, T.S. Pan, A.K. Dhingra
School of Mechanical Engineering
Purdue University, West Lafayette, IN 47907

V.B. Venkayya
WRDC/FIBRA, Wright-Patterson Air Force Base, OH 45433

and

V. Kumar
Corporate Research and Development
General Electric Co., Schenectady, NY 12301

Abstract

This paper presents the applicability of a biological model, based on genetic evolution, for engineering design optimization. Algorithms embodying the ideas of reproduction, crossover and mutation are developed and applied to solve different types of structural optimization problems. Both continuous and discrete variable optimization problems are solved. A two-bay truss for maximum fundamental frequency is considered to demonstrate the continuous variable case. The selection of locations of actuators in an actively controlled structure, for minimum energy dissipation, is considered to illustrate the discrete variable case.

Introduction

Over the years, numerous techniques have been developed for optimizing the design and performance of engineering systems. Despite the wide variety of available techniques, no one method has proved to be entirely satisfactory across the broad spectrum of problems confronting a design engineer.

Physical and biological systems are well known for their robustness, a balance between efficiency and efficacy necessary for survival under different environments. Features for self guidance, repair and reproduction are the rule in such systems, whereas they barely exist in most sophisticated artificial systems. Thus, where robust performance is required, nature does is better; the secrets of adaption and survival are best learned from a careful study of molecular evolution. The present work will investigate into the applicability of a biological model, based on genetic evolution, for engineering design optimization [1].

If one compares the biological optimization process with a mathematical optimization process, one can notice several similarities. The fitness function and chromosomes (genes) in a biological process are, respectively, similar to the objective function and design variables in a mathematical optimization process. Upon a close examination of strategies both the processes employ, one notices they have still more in common. The conventional strategy of mathematical optimization, iterative improvement, is just like evolution; its two elements, a

variation scheme and a decision rule are analogous to a crossover mechanism and a selection principle. All of these suggest pursuing the analogy between genetic evolution and mathematical optimization. It is expected that this will enable us to obtain a better algorithm for mathematical optimization.

In this work, algorithms embodying the ideas of reproduction, crossover and mutation have been developed and applied to solve different types of structural optimization problems. Both continuous and discrete variable optimization problems have been solved to demonstrate the efficiency of the algorithms.

Basic Procedure

A simple genetic algorithm involves copying strings and swapping partial strings between two mating strings. Three basic but very important operators - reproduction, crossover and mutation - are used to produce new generations over and over again. Reproduction is a randomized selection process in which individual strings are copied according to their fitness (or objective function) value. Similar to the natural selection process in which a Darwinian survival of the fittest among string creatures, the string with a higher fitness value has a higher probability to be reproduced. The probability of reproduction can be determined by dividing the individual fitness by the sum of fignesses of the current generation. Crossover (or recombination) is a primary operator in the mating process which generates the offsprings or new generations. Two steps are involved in the crossover process. The first step is randomly selecting the position to break (crossover) between two mating couple. The second step is reunion of these two mating couple. In natural genetic theory, the recombination fraction depends on the distance between the chromosomes of the mating couple. Mutation plays a secondary role in the operation of genetic algorithms. Mutation, changing a particular bit of coded string from 0 to 1 or vice versa, is a random walk through the string space.

Illustrative Examples

Example 1:

Design of a Truss for Maximum Natural Frequency

The ten-member planar truss shown in Fig. 1(a) is required to support the loads indicated. The objective is to maximize the fundamental frequency of vibration with constraints on member stresses. The design variables are the cross sectional areas of the members (continuous) with upper and lower bounds. The data are: E = Young's modulus = 10^7 psi, ρ = specific weight = 0.1 lb/in^3, permissible stress = 25000 psi, $x_i^{(\ell)}$ = lower bound on areas = 0.1 in^2, $x_i^{(u)}$ = upper bound on areas = 10.0 in^2, p_c = probability of crossover = 0.8, p_m = probability of mutation = 0.002, and population size = 50.

Figures 1(b) and 1(c) show the best-of-generation and average values of fitness indices for three different runs of the genetic algorithm. The solid line corresponds to the optimum results obtained using a gradient based search procedure. The three different runs gave values of 38.545 Hz, 39.459 Hz and 37.437 Hz for ω_1 which correspond to approximately 60% improvement over the value of 24.028 Hz given by a gradient based method. This large improvement is due to the non-dependence of genetic algorithms on gradient information, which in turn makes it less likely to get trapped in a local optimum.

Example 2:

Selection of Locations of Actuators

The problem of determination of locations of actuators and sensors in actively controlled structures is formulated as a zero - one programming problem. The equations of motion of a flexible structure can be expressed as [2,3]:

$$M \ddot{v} + C \dot{v} + K v = D u \tag{1}$$

where M,C,K and D are the mass, damping, stiffness and the input matrices and v and u are the displacement and the input vectors. Letting $x^T = \{v^T \ \dot{v}^T\}$, Eq. (1) can be rewritten as

$$\dot{x} = A x + B u \tag{2}$$

where

$$A = \begin{bmatrix} 0 & I \\ -M^{-1} K & -M^{-1} C \end{bmatrix} \tag{3}$$

and

$$B = \begin{bmatrix} 0 \\ M^{-1} D \end{bmatrix} \tag{4}$$

The optimal linear quadratic regulator method is applied to design the control gain of the feedback controller for simplicity. Accordingly, the input vector u is given by

$$u = - R^{-1} B^T P x \tag{5}$$

where P satisfies the matrix Riccati equation:

$$A^T P + P A - P B R^{-1} B^T P + Q = 0 \tag{6}$$

where Q is a positive semidefinite output weighting matrix and R is a positive definite input weighting matrix. Since the input matrix B is a function of the locations of the actuators, the system equations (1) and (2) will be changed if the locations of actuators are changed.

The inverse of the input weighting matrix R is assumed to be a diagonal matrix containing ones and zeros only with 1's corresponding to the locations with actuators and 0's corresponding to those without actuators. Hence, the system equations will remain unchanged during the optimization process. Similarly, the output weighting matrix Q can be modified such that the system equations for the estimation part are unchanged during the optimization process for sensor location selection. For simplicity, the estimation part is neglected, and only the actuator location selection problem is considered in this work.

The objective function (criterion) proposed to be used in the actuator location selection problem is the energy dissipated by the active controller which can be written as

$$E_c = \frac{1}{2} \int_0^\infty \dot{q}^T D_c \, \dot{q} \, dt \qquad (7)$$

where $\dot{q}$ is the velocity vector and D_c is the induced damping matrix by the active controller. Let

$$R^{-1} = \text{diag} \left[\bar{r}_1 \, \bar{r}_2 \cdots \bar{r}_n \right] \qquad (8)$$

denote the inverse weighting matrix, with $\bar{r}_i$ indicating a binary variable denoting presence or absence of an actuator at position i. Then the zero-one problem for the actuator location selection problem can be stated as follows:

$$\underset{\bar{r}_1, \bar{r}_2, \ldots, \bar{r}_n}{\text{maximize } E_c} \qquad (9)$$

subject to

$$\sum_{i=1}^{n} \bar{r}_i = m$$

and $\bar{r}_i = 0$ or 1 ; $i = 1, 2, ..., n$

For illustration, the six-bay truss shown in Fig. 3(a) is considered with 3 actuators. The initial population size and the maximum number of generations are chosen as 20 and 40, respectively. The results obtained with $p_c = 0.8$ and $p_m = 0.001$ are shown in Figs. 3(b) and 3(c). The global optimal solution with the actuators at positions (2,3,4) was found in less than 271 function evaluations.

Conclusion

Genetic algorithmic approaches are proposed for solving continuous and discrete variable optimization problems. These algorithms, although simple guided random search procedures, are found to be very effective in solving practical structural design problems. In most cases, the global optimal solution is found in reasonable number of function evaluations. Since genetic algorithms tend to find global optimal solutions, the results are often superior to those given by the gradient based optimization procedures.

Acknowledgements

References

1. D.E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning," Addison-Wesley, Reading, MA, 1989.

2. S.S. Rao, V.B. Venkayya and N.S. Khot, "Game Theory Approach for the Integrated Design of Controls and Structures," AIAA Journal, Vol. 26, pp. 463-469, 1988.

3. S.S. Rao, T.S. Pan and V.B. Venkayya, "Optimal Placement of Actuators in Actively Controlled Structures Using Genetic Algorithms," AIAA Journal, Vol. 28, 1990 (in press).
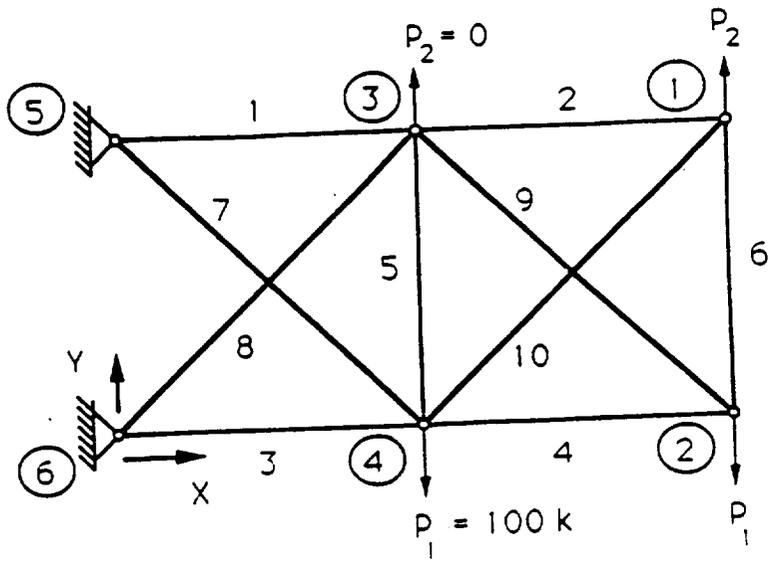
(a) Ten-Member Planar Truss
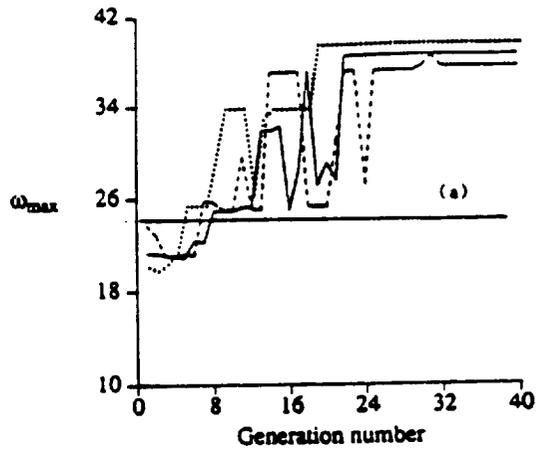
(b) Best-of-Generation Fitness Index
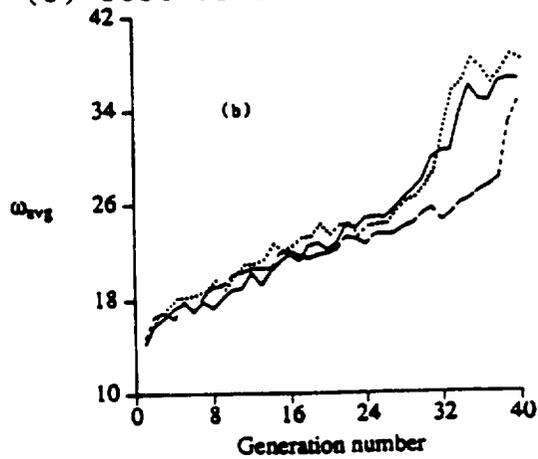
(c) Average Value of Fitness Index

Figure 1

322

(a) Six-Bay Truss

(b) Average Fitness Change with Different Initial Conditions

(c) Maximal Fitness Change with Different Initial Conditions

Figure 2

A HOMOTOPY ALGORITHM FOR SYNTHESIZING ROBUST CONTROLLERS FOR FLEXIBLE STRUCTURES VIA THE MAXIMUM ENTROPY DESIGN EQUATIONS

Emmanuel G. Collins, Jr. and Stephen Richter

Harris Corporation
Government Aerospace Systems Division
MS 22/4842
Melbourne, FL 32902

Abstract

One well known deficiency of LQG compensators is that they do not guarantee any measure of robustness. This deficiency is especially highlighted when considering control design for complex systems such as flexible structures. There has thus been a need to generalize LQG theory to incorporate robusness constraints. Here we describe the maximum entropy approach to robust control design for flexible structures, a generalization of LQG theory, pioneered by Hyland, which has proved useful in practice. The design equations consist of a set of coupled Riccati and Lyapunov equations. A homotopy algorithm that is used to solve these design equations is presented.

1. Introduction

The linear-quadratic-gaussian (LQG) compensator has been developed to facilitate the design of control laws for complex systems, i.e., systems that have large order and/or are multi-input multi-output. An LQG compensator minimizes a quadratic performance index and (under mild conditions) is guaranteed to yield an internally stable closed-loop system. Unfortunately, however, standard LQG theory does not guarantee any measure of robustness. This deficiency is especially highlighted when considering control design for systems with many lightly-damped modes such as flexible structures and has necessitated generalizations of LQG theory that incorporate practical robustness constraints for this class of systems.

In recent years there have been several extensions of LQG theory that have included some form of robustness constraints (e.g., [1-5]). Many of these results can be shown to be equivalent to enforcing the requirements of the small gain theorem for some system transfer function [6]. It is not difficult to show that controllers designed with these results will not generally yield high performance, robust controllers for flexible structures since they assume a very crude model of the uncertainty (i.e., complex as opposed to real parameter uncertainty) within the controller bandwidth. However, one result that is fundamentally different in nature is the maximum entropy approach to control design for flexible structures [1-3] pioneered by Hyland.

The maximum entropy approach is a generalization of LQG theory that explicitly allows the consideration of a form of real-valued parametric uncertainty. The name derives from its apparently coincidental relationship to certain stochastic models. Controllers designed using this method have several useful features. First of all, for certain structural control problems the maximum entropy controllers will be positive real in the controller bandwidth. This is extremely important for flexible structures since positive real controllers often allow high performance control even when there is significant system uncertainty. In addition, the maximum entropy controllers are often reduced-order controllers. Thus in practice the maximum entropy robustness constraint aids in choosing the order of the controller and is a numerical aid in controller reduction.

The maximum entropy design equations consist of a set of four equations, two Riccati equations coupled to two Lyapunov equations. The coupling between the equations is a function of the assumed uncertainty. If no uncertainty is assumed the four equations decouple and the design equations become the two standard LQG Riccati equations. To enable this theory for engineering practice it is of course important to develop

robust and efficient numerical algorithms for solving the design equations. This paper presents such an algorithm.

The algorithm described here is a homotopy algorithm. The advantage of a homotopy algorithm is that it is based on theory that is global in nature. In particular, the convergence of homotopy algorithms is generally not dependent upon having initial conditions which are "close" to the actual solution.

The paper is organized as follows. Section 2 briefly describes the general construction of both discrete and continuous homotopy algorithms. Section 3 presents the maximum entropy design equations and describes their derivation. Section 4 presents a flowchart describing the homotopy algorithm while Section 5 illustrates some features of maximum entropy controllers seen in control design for the ACES structure at NASA Marshall Space Flight Center. Finally Section 6 offers some concluding remarks.

2. A Brief Description of Homotopy Algorithm Development

A "homotopy" is a continuous deformation of one function into another. Over the past several years, homotopy or continuation methods (whose mathematical basis is algebraic topology and differential topology) have received significant attention in the mathematics literature and have been applied successfully to several important problems (e.g., [7–8]). Recently, the engineering literature has also begun to recognize the utility of these methods for engineering applications (see e.g., [9–12]). The purpose of this section is to provide a very brief description of homotopy methods for finding the solutions of nonlinear algebraic equations. The reader is referred to [9] for additional details.

The basic problem is as follows. Given sets U and V contained in $\mathbb{R}^n$ and a mapping $F\colon U \to V$, find solutions to

$$F(u) = 0. \tag{2.1}$$

Homotopy methods embed the problem (2.1) in a larger problem. In particular let $H\colon Y \times [0,1] \to \mathbb{R}^n$ be chosen so that (i) $H(u,1) = F(u)$, (ii) there exists a known solution u_0 to $H(\cdot,0) = 0$, (iii) there exists a continuous curve $(u(\gamma),\gamma)$ in $\mathbb{R}^n \times [0,1]$ such that $H(u(\gamma),\gamma) = 0$ for $\gamma \in [0,1]$ with $(u(0),0) = (u_0,0)$ and (iv) the curve $(u(\gamma),\gamma)$ is differentiable. A homotopy algorithm then constructs a procedure to compute the actual curve $(u(\gamma),\gamma)$ such that the initial solution $u(0)$ is transformed to a desired solution $u(1)$ satisfying $0 = H(u(1),1) = F(u(1))$.

There are two related methods of following the curve $(u(\gamma),\gamma)$, namely, continuous methods and discrete methods. Continuous methods work by differentiating $H(u(\gamma),\gamma) = 0$ with respect to γ obtain Davidenko's differential equation

$$\frac{\partial H}{\partial u}\frac{du}{d\gamma} + \frac{\partial H}{\partial \gamma} = 0. \tag{2.2}$$

Together with $u(0) = u_0$, (2.2) defines an initial value problem which by numerical integration from 0 to 1 yields the desired solution $u(1)$.

Discrete methods work by partitioning the interval $[0,1]$ to obtain a finite chain of problems

$$H(u,\gamma_k) = 0, \quad 0 = \gamma_0 < \gamma_1 < \cdots < \gamma_N = 1. \tag{2.3}$$

Starting with a known solution $u(0) = u_0$, $u(\gamma_{k+1})$ is computed by a local interation scheme with $u(\gamma_k)$ as the starting point. Although this approach avoids direct integration of Davidenko's equation, it does follow the homotopy path from $u(0)$ to $u(1)$.

The algorithm presented in Section 4 uses both discrete homotopy and continuous homotopy techniques. Globally, the algorithm is a discrete homotopy algorithm. However, the local iteration scheme which advances the discrete homotopy uses continuous homotopies to solve certain nonlinear algebraic equations which are subsets of the maximum entropy design equations described below.

325

3. The Maximum Entropy Design Equations

Consider the system

$$\dot{x}(t) = Ax(t) + Bu(t) + w_1(t), \quad y(t) = Cx(t) + w_2(t) \tag{3.1a, b}$$

where $x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}, y \in \mathbb{R}^{n_y}, w_1 \in \mathbb{R}^{n_x}$ is white disturbance noise with intensity $V_1 \geq 0$, and $w_2 \in \mathbb{R}^{n_y}$ is white observation noise with intensity $V_2 > 0$.

In the standard LQG problem we desire to design an nth order compensator,

$$\dot{x}_c(t) = A_c x_c(t) + B_c y(t), \quad u(t) = -C_c x_c(t) \tag{3.2a, b}$$

which minimizes the steady-state performance criterion

$$J(A_c, B_c, C_c) \triangleq \lim_{t \to \infty} E[x^T(t) R_1 x(t) + u^T(t) R_2 u(t)] \tag{3.3}$$

where $x_c \in \mathbb{R}^n, R_1 = R_1^T \geq 0$ and $R_2 = R_2^T > 0$.

Define $\tilde{A}$ and $\tilde{R}$ as

$$\tilde{A} \triangleq \begin{bmatrix} A & -BC_c \\ B_c C & A_c - B_c D C_c \end{bmatrix}, \quad \tilde{R} \triangleq \begin{bmatrix} R_1 & 0 \\ 0 & C_c^T R_2 C_c \end{bmatrix}. \tag{3.4, 5}$$

and let $\tilde{Q}$ be the closed-loop steady-state covariance, i.e.,

$$0 = \tilde{A}\tilde{Q} + \tilde{Q}\tilde{A}^T + \tilde{V}. \tag{3.6}$$

Then, the cost can be expressed as

$$J(A_c, B_c, C_c) = \text{tr } \tilde{Q}\tilde{R} \tag{3.7}$$

and the solution to the LQG problem can be obtained by using Lagrange multipliers to optimize (3.7) subject to the constraint (3.6).

The standard LQG equations do not explicitly incorporate any robustness constraints. Thus, it is important to find generalizations of these equations that allow the synthesis of robust controllers for flexible structures. One such generalization is the maximum entropy design equations, whose name derives from their coincidental relationship to stochastic modeling.

We now assume that the A matrix of (3.1) is in the form

$$A = \text{ block-diag}\{A^{(1)}, A^{(2)}\} \tag{3.8}$$

where $A^{(1)}$ represents the nominal dynamics of the uncertain modes and is in real normal form; for example,

$$A^{(1)} = \text{ block-diag}\left\{ \begin{bmatrix} -\nu_1 & \omega_1 \\ -\omega_1 & -\nu_1 \end{bmatrix}, -\nu_2, \begin{bmatrix} -\nu_3 & \omega_3 \\ -\omega_3 & -\nu_3 \end{bmatrix} \right\}. \tag{3.9}$$

Also, assume that the only the modes with complex eigenvalues (corresponding to the 2×2 blocks $\begin{bmatrix} -\nu_j & \omega_j \\ -\omega_j & -\nu_j \end{bmatrix}$) are uncertain and that the uncertainty patterns $A_i \in \mathbb{R}^{n_x \times n_x}$ are of the form

$$A_i = \text{ block-diag}\left\{ 0, \ldots, 0, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, 0, \ldots, 0 \right\}. \tag{3.10}$$

Notice that for lightly damped modes A_i essentially corresponds to frequency uncertainty. However, in practice this representation of uncertainty can also be used to account for uncertainties in the damping and mode shapes.

The ME equations are developed by modifying the constraint equation (3.6) as follows:

$$0 = \tilde{A}_s \tilde{Q} + \tilde{Q} \tilde{A}_s^T + \tilde{V} + \sum_{i=1}^{n_\alpha} \tilde{A}_i \tilde{Q} \tilde{A}_i^T \tag{3.11}$$

where n_α is the number of uncertain modes,

$$\tilde{A}_i = \text{block-diag}\{A_i, 0_n\}, \quad \tilde{A}_s = \tilde{A} + \frac{1}{2} \sum_{i=1}^{n_\alpha} \tilde{A}_i^2. \tag{3.12, 13}$$

Optimization, using the constraint equation (3.11) then yields the following set of design equations:

$$0 = A_s Q + Q A_s^T + V_{\hat{Q}} - Q C^T V_2^{-1} C Q + \sum_{i=1}^{n_\alpha} \alpha_i^2 A_i Q A_i^T \tag{3.14}$$

$$0 = A_s^T P + P A_s + R_{\hat{P}} - P B R_2^{-1} P B^T + \sum_{i=1}^{n_\alpha} \alpha_i^2 A_i P A_i^T \tag{3.15}$$

$$0 = A_P \hat{Q} + \hat{Q} A_P^T + Q C^T V_2^{-1} C Q \tag{3.16}$$

$$0 = A_Q^T \hat{P} + \hat{P} A_Q + P B R_2^{-1} B^T P \tag{3.17}$$

where

$$A_s = A + \sum_{i=1}^{n_\alpha} \alpha_i^2 A_i^2 \tag{3.18}$$

$$V_{\hat{Q}} = V_1 + \sum_{i=1}^{n_\alpha} \alpha_i^2 A_i \hat{Q} A_i^T, \quad R_{\hat{P}} = R_1 + \sum_{i=1}^{n_\alpha} \alpha_i^2 A_i^T \hat{P} A_i \tag{3.19}$$

$$A_P = A_s - B R_2^{-1} B^T P, \quad A_Q = A_s - Q C^T V_2^{-1} C. \tag{3.20}$$

The controller gains are given by

$$A_c = A - B R_2^{-1} B^T P - Q C^T V_2^{-1} C, \quad B_c = Q C^T V_2^{-1}, \quad C_c = R_2^{-1} B^T P. \tag{3.21}$$

Notice that if $\alpha_i \triangleq 0$ then (3.14) and (3.15) decouple from (3.16) and (3.17) and become the LQG Riccati equations.

4. A Homotopy Algorithm for Solving the Maximum Entropy Design Equations

Let $\gamma \in [0, 1]$ denote the homotopy parameter and consider the following equations which are used to define the homotopy algorithm:

$$0 = A_s(\gamma) Q^{(k+1)} + Q^{(k+1)} A_s^T(\gamma) + V_{\hat{Q}}^{(k)}(\gamma) - Q^{(k+1)} C^T V_2^{-1} C Q^{(k+1)} + \gamma \sum_{i=1}^{n_\alpha} \alpha_i^2 A_i Q^{(k+1)} A_i^T \tag{4.1}$$

$$0 = A_s^T(\gamma) P^{(k+1)} + P^{(k+1)} A_s(\gamma) + R_{\hat{P}}^{(k)}(\gamma) - P^{(k+1)} B^T R_2^{-1} B P^{(k+1)} + \gamma \sum_{i=1}^{n_\alpha} \alpha_i^2 A_i^T P^{(k+1)} A_i \tag{4.2}$$

$$0 = A_P^{(k+1)}(\gamma) \hat{Q}^{(k+1)} + \hat{Q}^{(k+1)} A_P^{(k+1)T}(\gamma) + Q^{(k+1)} C^T V_2^{-1} C Q^{(k+1)} \tag{4.3}$$

$$0 = A_Q^{(k+1)T}(\gamma) \hat{P}^{(k+1)} + \hat{P}^{(k+1)} A_Q^{(k+1)}(\gamma) + P^{(k+1)} B R_2^{-1} B P^{(k+1)}. \tag{4.4}$$

where

$$A_s(\gamma) = A + \gamma \sum_{i=1}^{n_\alpha} \alpha_i A_i^2 \tag{4.5}$$

$$V_{\hat{Q}}^{(k)}(\gamma) = V_1 + \gamma \sum_{i=1}^{n_a} \alpha_i^2 A_i \hat{Q}^{(k)} A_i^{\mathrm{T}}, \quad R_{\hat{P}}^{(k)}(\gamma) = R_1 + \gamma \sum_{i=1}^{n_a} \alpha_i^2 A_i^{\mathrm{T}} \hat{P}^{(k)} A_i \tag{4.6}$$

$$A_P^{(k)}(\gamma) = A_s(\gamma) - B R_2^{-1} B^{\mathrm{T}} P^{(k)}, \quad A_Q^{(k)}(\gamma) = A_s(\gamma) - Q^{(k)} C^{\mathrm{T}} V_2^{-1} C. \tag{4.7}$$

Also, define

$$\Delta_Q^{(k)}(\gamma) \triangleq A_s(\gamma) Q^{(k)} + Q^{(k)} A_s^{\mathrm{T}}(\gamma) + V_{\hat{Q}}^{(k)}(\gamma) - Q^{(k)} C^{\mathrm{T}} V_2^{-1} C Q^{(k)} + \gamma \sum_{i=1}^{n_a} \alpha_i^2 A_i Q^{(k)} A_i^{\mathrm{T}} \tag{4.8}$$

$$\Delta_P^{(k)}(\gamma) \triangleq A_s^{\mathrm{T}}(\gamma) P^{(k)} + P^{(k)} A_s(\gamma) R_{\hat{P}}^{(k)}(\gamma) - P^{(k)} B^{\mathrm{T}} R_2^{-1} B P^{(k)} + \gamma \sum_{i=1}^{n_a} \alpha_i^2 A_i^{\mathrm{T}} P^{(k)} A_i \tag{4.9}$$

$$\Delta_{\hat{Q}}^{(k)}(\gamma) \triangleq A_P^{(k)}(\gamma) \hat{Q}^{(k)} + \hat{Q}^{(k)} A_P^{(k)^{\mathrm{T}}}(\gamma) + Q^{(k)} C^{\mathrm{T}} V_2^{-1} C Q^{(k)} \tag{4.10}$$

$$\Delta_{\hat{P}}^{(k)}(\gamma) \triangleq A_Q^{(k)^{\mathrm{T}}}(\gamma) \hat{P}^{(k)} + \hat{P}^{(k)} A_Q^{(k)}(\gamma) + P^{(k)} B R_2^{-1} B^{\mathrm{T}} P^{(k)}. \tag{4.11}$$

Equations (4.1)–(4.4) are identical in form to (3.14)–(3.17). In the homotopy algorithm the matrix functions $\Delta_Q^{(k)}(\gamma)$, $\Delta_P^{(k)}(\gamma)$, $\Delta_{\hat{Q}}^{(k)}(\gamma)$ and $\Delta_{\hat{P}}^{(k)}(\gamma)$ defined respectively by (4.8)–(4.11) are considered equation errors.

The normalized norms of the equation errors are defined by

$$e_Q^{(k)} \triangleq \|\Delta_Q^{(k)}(\gamma)\|_A / \|V_{\hat{Q}}(\gamma)\|_A, \quad e_P^{(k)} \triangleq \|\Delta_P^{(k)}(\gamma)\|_A / \|R_{\hat{P}}(\gamma)\|_A \tag{4.12}$$

$$e_{\hat{Q}}^{(k)} \triangleq \|\Delta_{\hat{Q}}^{(k)}(\gamma)\|_A / \|Q^{(k)} C^{\mathrm{T}} V_2^{-1} C Q^{(k)}\|_A, \quad e_{\hat{P}}^{(k)} \triangleq \|\Delta_{\hat{P}}^{(k)}(\gamma)\|_A / \|P^{(k)} B R_2^{-1} B^{\mathrm{T}} P^{(k)}\|_A \tag{4.13}$$

where

$$\|M\|_A \triangleq \max_{i,j} |m_{i,j}|. \tag{4.14}$$

The maximum normalized error norm is denoted by $e_{\max}^{(k)}(\gamma)$ and defined by

$$e_{\max}^{(k)}(\gamma) \triangleq \max\{e_Q^{(k)}(\gamma), e_P^{(k)}(\gamma), e_{\hat{Q}}^{(k)}(\gamma), e_{\hat{P}}^{(k)}(\gamma)\}. \tag{4.15}$$

Figure 4.1 presents a general flowchart of the homotopy algorithm. The outer (j) loop corresponds to a discrete homotopy. The discrete homotopy is advanced by the iteration scheme described by the inner (j) loop. The inner loop requires the solution of (4.1) and (4.2) using continuous homotopies. The integration schemes we have implemented to advance the continuous homotopy use Euler integration with a Newton correction.

The algorithm assumes that for some positive integer N

$$0 = \gamma_0 < \gamma_1 < \cdots < \gamma_{N-1} < \gamma_N = 1 \tag{4.16}$$

The sequence $\{\varepsilon_j\}_{j=0}^{N}$ of "small" positive numbers determines how closely the algorithm actually tracks the homotopy curve.

5. Illustration of Maximum Entropy Controllers

This section illustrates some important features of maximum entropy (ME) controllers. The results presented here were developed while designing decentralized control laws for the ACES structure located at NASA Marshall Space Flight Center [13]. This testbed has been used to conduct experiments on structural control and is especially suited for studying line-of-sight (LOS) control issues.

328

Figure 5.1 shows one of the two major loops used to design vibrational control laws. The second loop (not shown) has essentially the same features. Models of these loops were developed by using the Eigensystem Realisation Algorithm [14–15]. In this figure AGS-Y represents a torque actuator and BGYRO-Y represents an angular rate gyro sensor which is approximately colocated with the torque actuator. In the control design for each loop we penalised only the modes less than 3 Hz since these modes dominated the LOS performance. Due to significant uncertainty in the design models the ME robustification proved to be essential in designing high performance control laws to achieve the LOS objectives.

Figures 5.2 thru 5.4 demonstrate some of the key features of the ME controllers. Figure 5.2 shows the phase of the LQG and ME controllers in the controller bandwidth. While the LQG controllers phase varies widely the ME controllers became positive real in this region tending toward rate feedback as the uncertainty was increased. Thus, the ME designs provided the necessary stability robustness in the controller bandwidth.

Figure 5.3 shows the magnitude of the LQG and ME controllers in the controller bandwidth. Notice that the ME compensator magnitudes are much smoother than those of the LQG controllers, thus providing performance robustness. Another important implication is that the ME designs yielded robust controllers that are effectively reduced order controllers. In practice the full-order ME designs actually provide insight into the choice of the order of the compensator and is a numerical aid in synthesising reduced order controllers.

The higher authority controllers notched the high frequency modes that had large peak magnitude frequency responses. As illustrated by Figure 5.4, ME design was able to robustify the controller notches. That is, the controller notches were increased in both width and depth, thus providing robust gain stabilisation.

6. Concluding Remarks

This paper has described the maximum entropy design equations for robust control of flexible structures and has presented a homotopy algorithm to solve these equations. The iterative scheme used to advance the discrete homotopy parameter γ uses continuous homotopy methods to solve a certain pair of modified Riccati equations. The capabilities of the algorithm and some important features of the maximum entropy designs were illustrated by studying control design for the ACES Structure at NASA Marshall Space Flight Center.

References

1. D. C. Hyland, "Optimal Regulation of Structural Systems with Uncertain Parameters," MIT Lincoln Laboratory, TR-551, 2 February 1981, DDC#ADA-099111/7.

2. D. C. Hyland, "Maximum Entropy Stochastic Approach to Controller Design for Uncertain Structural Systems," *Proceedings of the American Control Conf.* pp. 680–699, Arlington, VA, June 1982.

3. D. S. Bernstein and D. C. Hyland, "The Optimal Projection/Maximum Entropy Approach to Designing Low-Order, Robust Controllers for Flexible Structures," *Proceedings of the IEEE Conference on Decision and Control*, pp. 745–752, Fort Lauderdale, FL, December 1985.

4. S. S. L. Chang and T. K. C. Peng, "Adaptive Guaranteed Cost Control of Systems with Uncertain Parameters," *IEEE Trans. Autom. Contr.*, Vol. AC–17, pp. 474–483, 1972.

5. D. S. Bernstein and W. M. Haddad, "The Optimal Projection Equations with Petersen Hollot Bounds: Robust Stability and Performance via Fixed Order Dynamic Compensation for Systems with Structured Real Parameter Uncertainty," *IEEE Trans. on Autom. Contr.*, Vol. AC–33, pp. 578–582, 1988.

6. A. Packard and J. Doyle, "Quadratic Stability with Real and Complex Perturbations," *IEEE Trans. Autom. Contr.*, Vol. 3, March 1990, to appear.

7. J. H. Avila, "The Feasibility of Continuation Methods for Nonlinear Equations," *SIAM J. Numer. Anal.*, Vol. 11, pp. 102–122, 1974.

8. L. T. Watson, "Numerical Linear Algebra Aspects of Globally Convergent Homotopy Methods," *SIAM Rev.*, Vol. 28, pp. 529–545, 1986.

9. S. L. Richter and R. A. DeCarlo, "Continuation Methods: Theory and Applications," *IEEE Trans. Circ. Syst.* Vol. CAS–30, pp. 347–352, 1983.

10. P. T. Kabamba, R. W. Longman and S. Jian-Guo, "A Homotopy Approach to the Feedback Stabilization of Linear Systems," *J. Guid. Control Dynamics*, Vol. 10, pp. 422–432, 1987.

11. Y. S. Shin, R. T. Haftka, L. T. Watson and R. H. Plautt, "Tracking Structural Optima as a Function of Available Resources by a Homotopy Method," *Computer Methods in Applied Mechanics and Engineering*, Vol. 70, pp. 151–164, 1988.

12. S. Richter and E. G. Collins, Jr., "A Homotopy Algorithm for Reduced-Order Controller Design Using the Optimal Projection Equations," *Proc. IEEE Conf. Dec. Contr.*, pp. 506–511, Tampa, FL, December 1989.

13. E. G. Collins, Jr., D. J. Phillips and D. C. Hyland, "Design and Implementation of Robust Decentralized Control Laws for the ACES Structure at the Marshall Space Flight Center," *Amer. Contr. Conf.*, pp. 1449–1454, San Diego, CA, May 1990.

14. J. N. Juang and R. S. Pappa, "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction," *J. Guid. Contr. Dyn.* Vol. 8, pp. 620–627, 1985.

15. J. N. Juang and R. S. Pappa, "Effects of Noise on Modal Parameters Identified by the Eigensystem Realization Algorithm," *J. Guid. Contr. Dyn.*, Vol. 9, pp. 294–303, 1986.

Intialize $j = 0$, $\gamma = \gamma_0 = 0$.

Initialize $Q^{(0)}, P^{(0)}, \hat{Q}^{(0)}$ and $\hat{P}^{(0)}$ by solving the LQG problem.

$k = 0$

Use (4.8)-(4.15) to compute the maximum normalised error norm $e_{\max}^{(k)}(\gamma)$.

$e_{\max}^{(k)}(\gamma) \leq \varepsilon_b$?

yes

no

Use a continuous homotopy to deform $Q^{(k)}$ and $P^{(k)}$ into the solutions $Q^{(k+1)}$ and $P^{(k+1)}$ of (4.1) and (4.2).

Use a standard Lyapunov solver to find the solutions $\hat{Q}^{(k+1)}$ and $\hat{P}^{(k+1)}$ of (4.3) and (4.4).

$k \leftarrow k + 1$

$Q^{(0)} \leftarrow Q^{(k)}$, $P^{(0)} \leftarrow P^{(k)}$
$\hat{Q}^{(0)} \leftarrow \hat{Q}^{(k)}$, $\hat{P}^{(0)} \leftarrow \hat{P}^{(k)}$

$\gamma = \gamma_j$

$j \leftarrow j + 1$

$\gamma = 1$?

no

yes

$Q = Q^{(k)}$, $P = P^{(k)}$
$\hat{Q} = \hat{Q}^{(k)}$, $\hat{P} = \hat{P}^{(k)}$

END
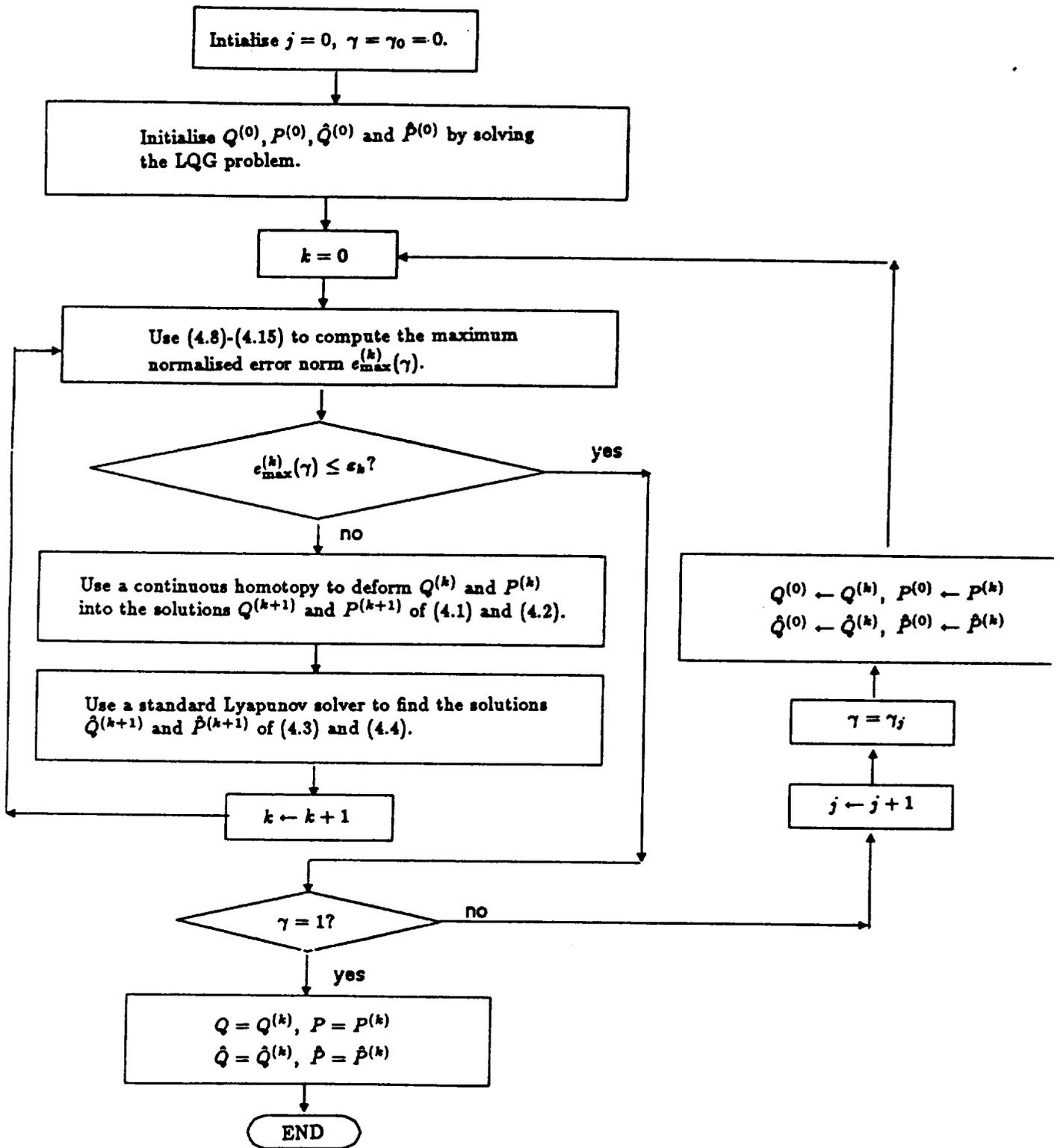
Figure 4.1. Flowchart of the Homotopy Algorithm for Solving the Maximum Entropy Design Equations.

AGS-Y TO BGYRO-Y
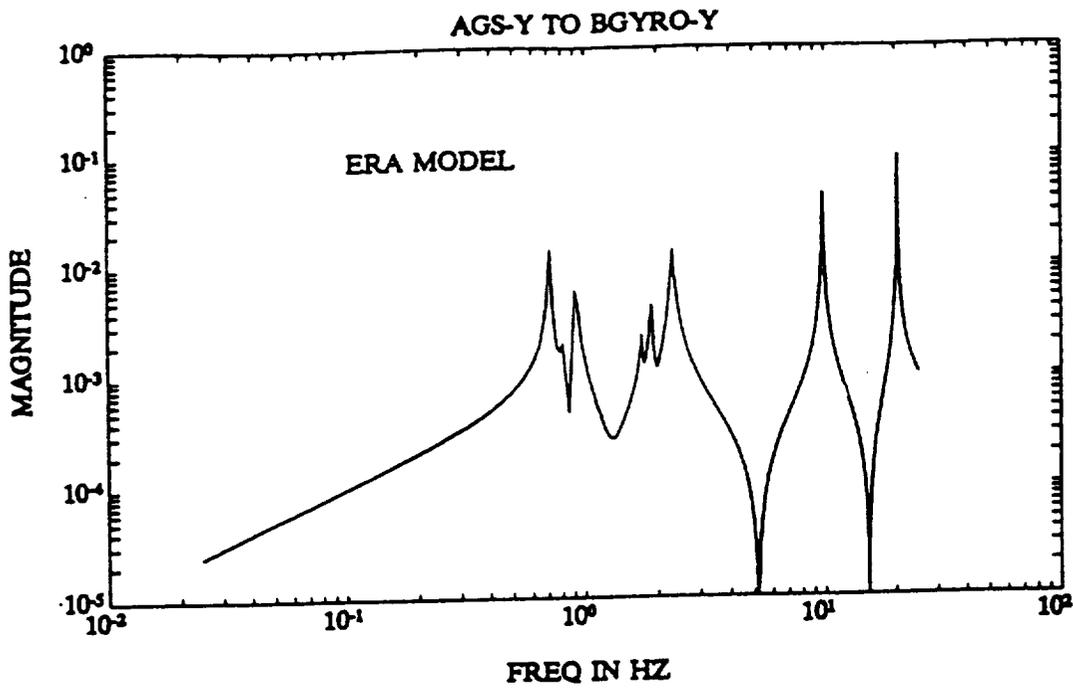
Figure 5.1. One of the two major feedback loops used to design vibration control laws for the ACES Structure at NASA Marshall Space Flight Center.

COMPENSATOR PHASE IN THE PERFORMANCE REGION

SOLID= W/OUT MAXIMUM ENTROPY

DASHED= WITH MAXIMUM ENTROPY
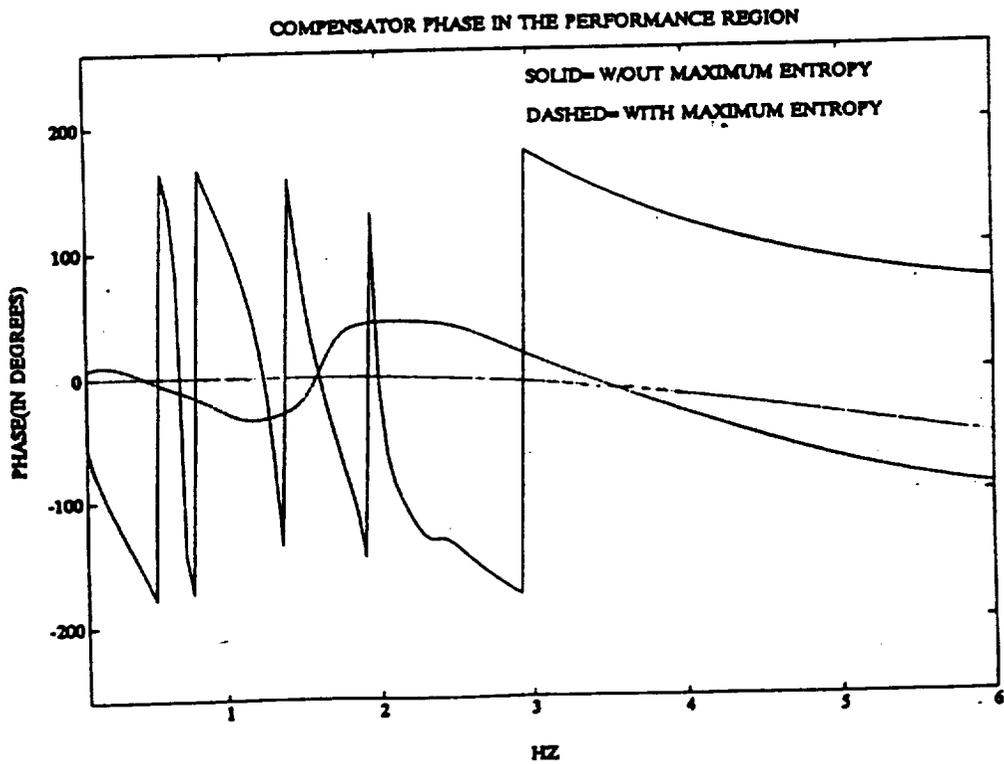
Figure 5.2. Maximum Entropy design rendered the compensators for the AGS to BGYRO loops positive real in the performance region.
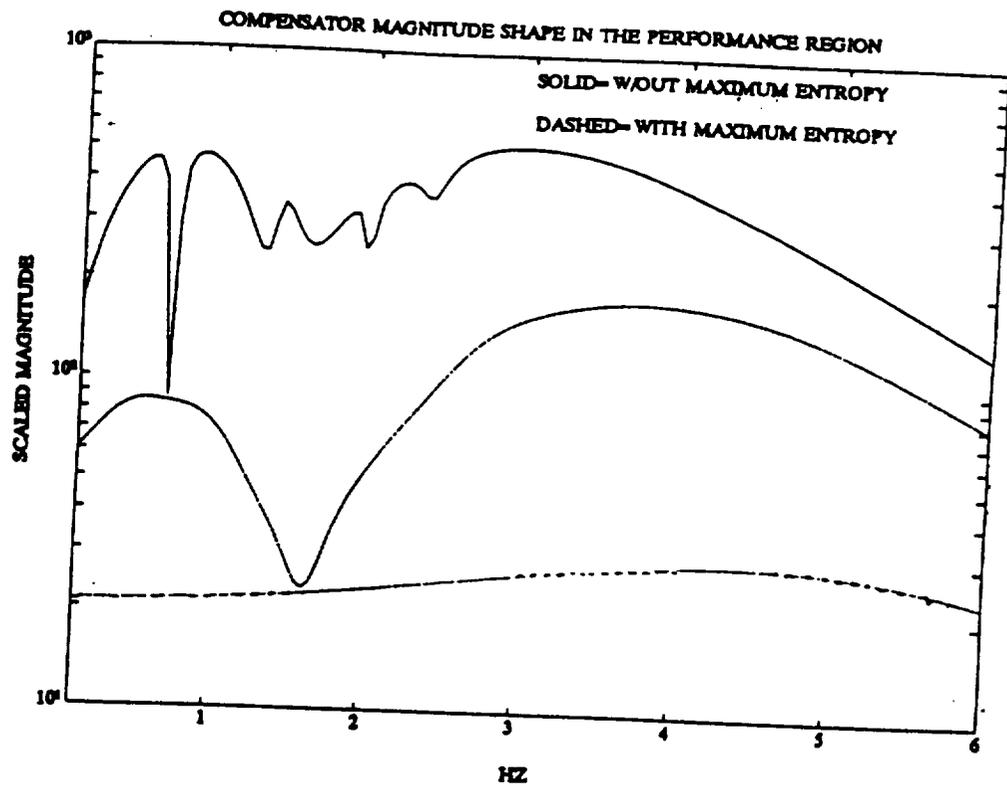
Figure 5.3. For the AGS to BGYRO loops Maximum Entropy design smoothed out the compensator magnitudes in the performance region, thus providing performance robustness and also indicating that the robust controllers were effectively reduced-order controllers.
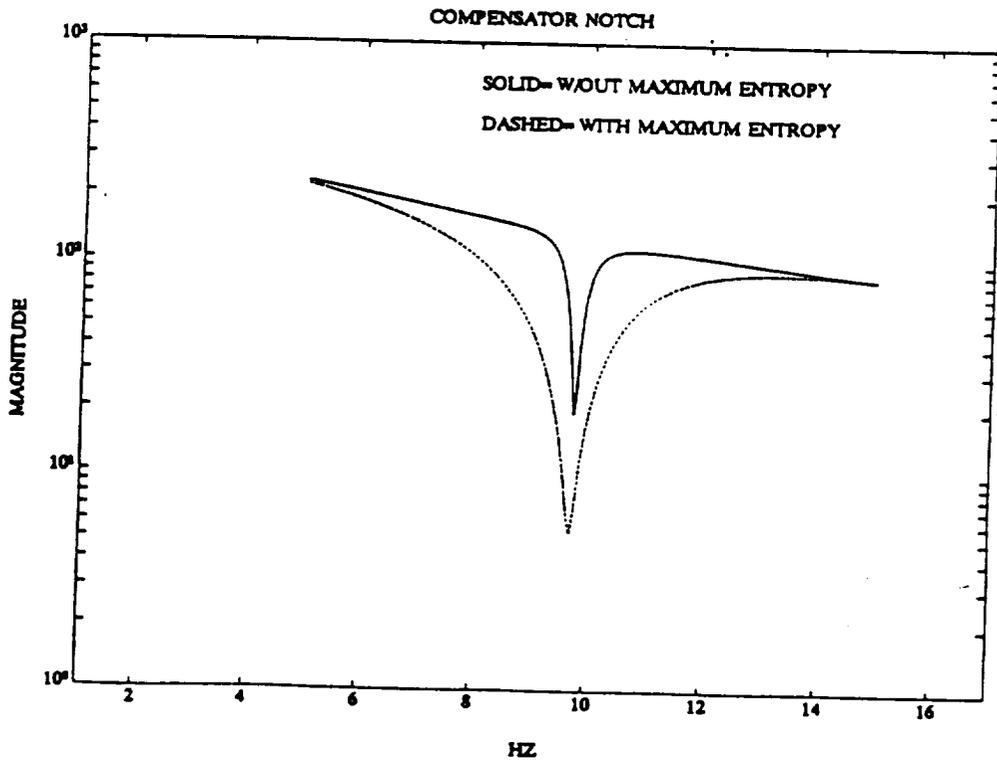
Figure 5.4. For the AGS to BGYRO loops Maximum Entropy design robustified the notches for the high frequency modes by increasing their width and depth.

CONSTRAINED OPTIMIZATION USING DESIGN OF EXPERIMENT SURFACES

Marvin Vance Bolt
General Dynamics - Fort Worth Division
Fort Worth, Texas

Abstract

An algorithm for solving constrained optimization problems is presented. First, design of experiment techniques are used to survey the design space. After evaluating the objective and constraint functions, as specified by Taguchi orthogonal arrays, analytical models of these functions are generated using a least-squares regression analysis. Next, a nonlinear programming package is used to optimize the analytical model. Based on the optimization information, the design space is reduced so as to close in around the minimum, and the entire procedure is repeated until convergence. An important feature of the algorithm is that function gradients are not required, therefore, for problems in which gradients would have to be estimated using finite-differences the number of function evaluations required for the optimization is significantly reduced, when compared with traditional nonlinear programming techniques. In addition, there is no requirement that the gradients must be smooth and continuous.

Introduction

Nonlinear programming techniques provide the solution to the following optimization problem:

$$\text{minimize } F(X)$$
$$\text{subject to } XL \leq X \leq XU \text{ and } CL \leq C(X) \leq CU.$$

$F(X)$ is the nonlinear objective function, X is the vector of design variables, and $C(X)$ is the vector of nonlinear constraints on X. The lower and upper bounds on X are XL and XU, respectively. Likewise, the lower and upper bounds on $C(X)$ are CL and CU, respectively. $F(X)$, $C(X)$, and their derivatives must be smooth and continuous.

NPSOL (Ref. 1) and ADS (Ref. 2) are two optimization routines that have successfully solved the stated problem, but the dependence of these algorithms on constraint and objective function gradients reduces the efficiency of the algorithms. Objective and constraint functions are often generated numerically by computationally intensive analysis programs, resulting in the need for finite-difference estimations of the gradients. The use of finite-differencing increases the number of function evaluations required for the optimization, making some problems prohibitively expensive. The requirement of smooth and continuous derivatives is another limitation. When data is loaded into computers in the form of tables, linear interpolation between points in the tables results in piecewise linear functions. This adversely affects the convergence of search methods that are dependent on gradients.

A scheme has been developed to address these limitations. Statistically based, design of experiment (DOE) techniques and response surface methodology (RSM) were combined to create an interface between a nonlinear programming algorithm and the optimization problem. The complete package of subroutines that was developed is designated CODES (Constrained Optimization using Design of Experiment Surfaces). Commercial software is available that uses DOE methods and RSM for optimization in an interactive environment (Ref 3). CODES differs from the commercial software in that it allows the optimization process to be fully automated (i.e. CODES can be linked directly to any analysis program). In addition, unlike the commercial software, CODES' iterative methods attempt to yield a solution that has been verified as a local optimum.

Experimental Design

CODES employs DOE methods to determine which points in the design space to use when performing the regression. The goal is to use the smallest number of points that will produce the best regression equation (indicated by the correlation coefficient). Experience has shown that D-Optimal designs (reference 4) best satisfy these needs, but the inability to create D-Optimal designs for large problems (greater than 8 variables) led to the use of Taguchi orthogonal arrays (Ref.5).

Response Surface Methodology

The objective and constraint functions are modeled using nonlinear least-squares regression. The regression equations are of the form,

$$F = \beta_1 + \beta_2 X_1 + \beta_3 X_1^2 + \beta_4 X_2 + \beta_5 X_1 X_2 + \beta_6 X_2^2 + \beta_7 X_3 + \beta_8 X_1 X_3 + \beta_9 X_2 X_3 + \beta_{10} X_3^2 + \ldots \quad (1)$$

and are generated as follows:

Define,

p = the number of data points (function evaluations) specified by the experimental design array

t = the number of terms in equation (1)

v = the number of design variables

X_{ij} = the value of the ith variable for the jth data point (by definition $X_{0j} = X_{i0} = 1$)

Y_j = the value of the objective (constraint) function(s) for the jth data point.

The solution, $\{\beta\}$, of the linear system,

$$[L]\{\beta\} = \{R\}$$

is the vector of coefficients from equation (1).

Matrix [L] is of size txt and vector $\{R\}$ is of length t. They are generated as follows:

$$L_{qr} = L_{rq} = \sum_{j=1}^{p} M_{qj} M_{rj} \qquad R_q = \sum_{j=1}^{p} M_{qj} Y_j \qquad (q=1,2,\ldots,t; \quad r=1,2,\ldots,t)$$

335

where [M] is a matrix of size txp and is given by,

$$M_{1j} = X_{0j}X_{0j'}$$
$$M_{2j} = X_{0j}X_{1j'} \qquad M_{3j} = X_{1j}X_{1j'}$$
$$M_{4j} = X_{0j}X_{2j'} \qquad M_{5j} = X_{1j}X_{2j'} \qquad M_{6j} = X_{2j}X_{2j'} \tag{2}$$
$$\ldots$$
$$M_{kj} = X_{0j}X_{vj'} \qquad M_{k+1j} = X_{1j}X_{vj'} \quad \ldots , \qquad M_{tj} = X_{vj}X_{vj} \quad (j=1,2,\ldots,p)$$

CODES was developed so that only the interaction terms (such as $\beta_5 X_1 X_2$) that are specified by the user are included in the regression. For example, if the term $\beta_5 X_1 X_2$ was excluded from equation (1), M_{5j} would assume the value of M_{6j} (equations(2)), M_{6j} becomes M_{7j}, etc.

Optimization

A collection of nonlinear programming subroutines is used to minimize and maximize the analytical representation of the problem. The actual objective and constraint functions are then evaluated at the optimum points of the regression equation. This is done in order to verify that the minimum (maximum) of the analytical model decreases (increases) the actual objective function when compared to the points used in the regression. The verification is also used to ensure that the actual constraints are satisfied. If, during the search, a feasible point has not yet been found, the point that minimizes the largest constraint violation is used as the current estimate of the minimum.

Reducing the Design Space

The design space is reduced by decreasing the search range for each variable. The goal is to close in around the optimum as quickly as possible in order to speed up convergence, although, reducing the search ranges too quickly increases the chances of eliminating the true minimum from the design space under consideration. This elimination can occur because the analytical model does not exactly represent the actual problem.

The strategy depicted in Figure 1 shows the design space reduction process. It consists of two primary steps:

(1) Reduce the search range for each variable so that the current estimate of the minimum is at the center of the new design space.
(2) If necessary, move the limits so they are within the previous search range.

The lower bounds on the variables X_1 and X_2 are $XL_1(1)$ and $XL_2(1)$, respectively (Figure 1a). Likewise, the upper bounds on the variables are $XU_1(1)$ and $XU_2(1)$. The search range for X_1 is changed by moving the lower limit to $XL_1(2)$ (Figure 1b). $XL_1(2)$ is found by bisecting X_{1Fmax} and X_{1Fmin} and $XU_1(2)$ is chosen so that X_{1Fmin} is at the midpoint of the new range. In order to ensure the search range is reduced, the upper limit of X_1 is reduced to $XU_1(3)$, the initial bound on X_1 (Figure 1c). The search range for X_2 is reduced in a similar fashion except $XU_2(2)$ is chosen to bisect X_{2Fmin} and $XU_2(1)$. This particular technique was chosen after trying several schemes, although it is possible there is a more efficient method.
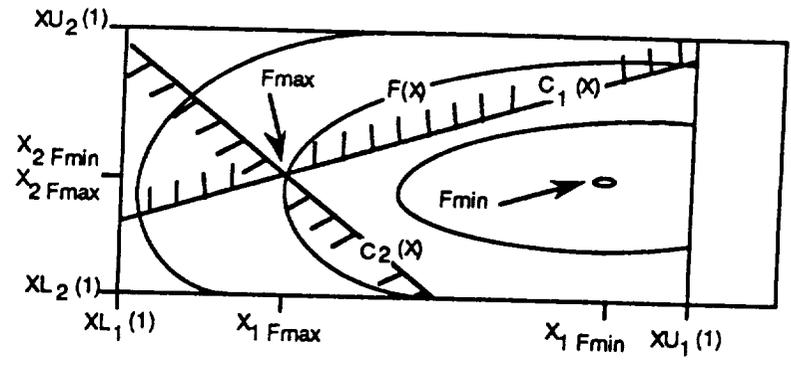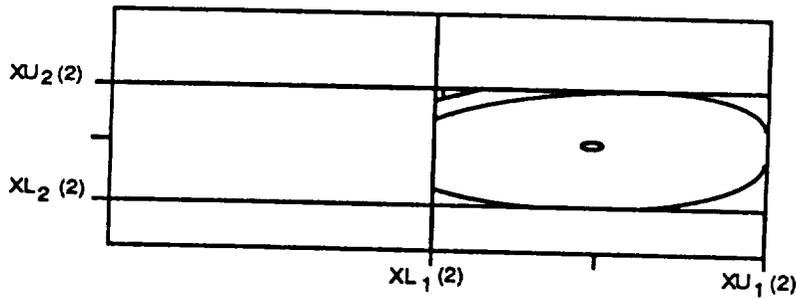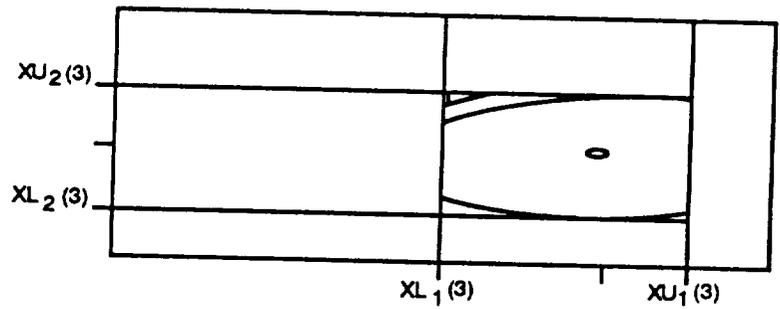
Figure 1 - Design space reduction process.

The CODES Algorithm

Figure 2 shows how all of the numerical procedures are integrated to form the optimization algorithm. If after two successive iterations the minimization of the analytical model yields the same results, within a specified tolerance, convergence is assumed. Analysis of the final search range for each variable indicates the validity of the solution. If the value of a variable is driven to the boundary of its' current search range, it is assumed that the minimum of the function was eliminated from the design space.
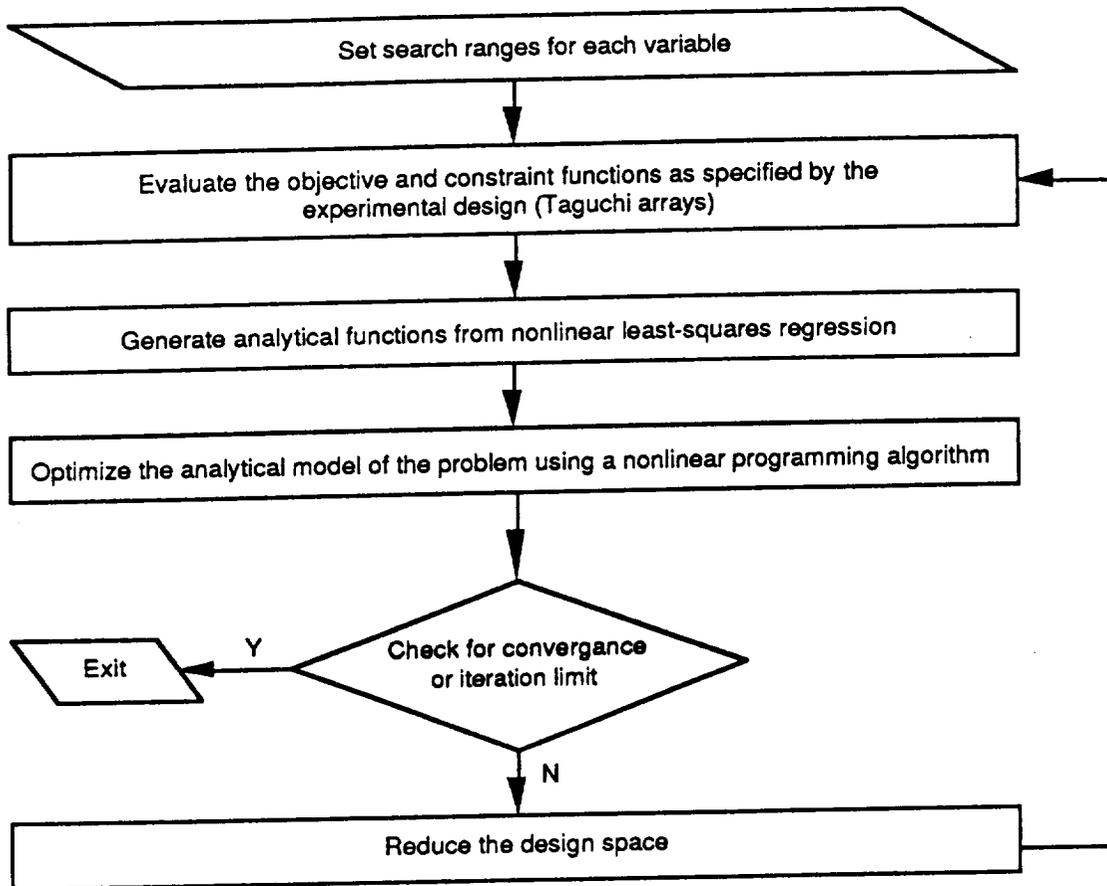
Figure 2 - CODES flow chart.

The availability of analytical equations that represent the problem is another advantage of CODES. Users of optimization software will (and should) always be suspicious of solutions generated by a computer. Analytical equations allow for greater visualization of the problem through computer graphics. This builds the user's confidence in the solution. In addition, the analytical models can be used for sensitivity studies at very little cost in computer time. A measure of the sensitivity of the objective and constraint functions is often just as important as the final solution.

Optimization Technique	Objective Function	Design Variables*					Number of Function Evaluations
	Take-off Gross Weight (lbs)	Wing Area (ft^2)	Engine Scale Factor	Aspect Ratio	Wing Thickness (% Chord)	Leading Edge Sweep (deg)	
Traditional Methods **	32,834	379	1.16	2.80	3.5	42	364
CODES	32,752	385	1.15	2.79	3.5	41	58

* Subject to 9 performance constraints
** Not directly aided by optimization software

Table 1 - A typical aircraft design synthesis parametric study.

Table 1 shows the potential savings that can be realized by implementing automated optimization techniques. Note that the same answer was found with both techniques, but the use of CODES resulted in an excess of 80% time savings. A final advantage of the methods presented here is CODES' potential to find global optima. This characteristic can best be understood by considering how DOE techniques evaluate points throughout the design space. In addition, performing the regression using a second order equation ensures that the mathematical model of the problem does not have many local optima.

Conclusions

Major savings can be realized when solving multidimensional optimization problems using the procedures presented here. The savings stem from shorter computer run time along with reduced manpower requirements to analyze and verify the results. The only significant limitation encountered so far is the maximum number of variable that can be considered using these methods. Preliminary indications place this limit in the range of 30 to 40 variables with the limiting factor being the size of available experimental design arrays.

References

1.) Gill P.E. et al.; *User's Guide for NPSOL (Version 4.0); A FORTRAN Package for Nonlinear Programming;* Report No. SOL-86-2; Dept. of Operations Research, Stanford University, California, Jan. 1986.

2.) Vanderplaats, G. N.; ADS - *A Fortran Program for Automated Design Synthesis*; NASA CR - No. 172460, 1984.

3.) BBN Software Products Corp.; *RS/1, RS/Explore, and RS/Discover User's Guides*, DEC. 1988.

4.) Federov, V. V.; *Theory of Optimal Experiments*, 1972; New York: Academic Press.

5.) Taguchi, G. *Introduction to Quality Engineering: Designing Quality into Products and Processes*; Asian Productivity Organization, White Plains, N.Y.; Kraus International Publications.

6.) Box, G. E. P. and Draper, Norman R.; *Empirical Model-Building and Response Surfaces*, 1987; N. Y. : Wiley.

7.) Draper, N. and Smith, H.; *Applied Regression Analysis*, 1981; N. Y.: Wiley.

8.) Galil, Z. and Kiefer, J.; "Time and Space-Saving Computer Methods, Related to Mitchell's DETMAX, For Finding D-Optimum Designs," 1980; *Technometrics*, 21:301-313.

9.) Jones, B. A. ; *A Robust Approach to Taguchi Methods*, 1988; Proceedings of the 12th Annual Rocky Mountain Quality Conference, pp. 325-339.

A UNIFIED APPROACH TO THE ANALYSIS AND DESIGN OF ELASTO-PLASTIC STRUCTURES WITH MECHANICAL CONTACT

Martin P. Bendsøe
Mathematical Institute
The Technical University of Denmark
DK-2800 Lyngby, Denmark

Niels Olhoff
Institute of Mechanical Engineering
Aalborg University
DK-9220 Aalborg East, Denmark

John E. Taylor
Department of Aerospace Engineering
The University of Michigan
Ann Arbor, MI 48109-2140, USA.

Abstract: With structural design in mind, a new unified variational model has been developed which represents the mechanics of deformation elasto-plasticity with unilateral contact conditions. For a design problem formulated as maximization of the load carrying capacity of a structure under certain constraints, the unified model allows for a simultaneous analysis and design synthesis for a whole range of mechanical behaviour.

1. INTRODUCTION. In order to express structural optimization, it is necessary to have an appropriate expression in variational form for the related structural mechanics analysis. With structural design in mind, a new unified variational model is presented which represents the mechanics of deformation elasto-plasticity with contact boundary conditions. The basic formulation of such problems is most natural in the form of a mixed variational model, with structural state expressed in terms of (independent) stress and displacement fields. However, a pure stress (force) method can also be obtained, and this constitutes the basis for the unified method developed for the formulation of a load carrying capacity design problem.

This paper constitutes an extension of earlier work on a unified model of elasto-plasticity that encompasses pure elasticity, elasto-plasticity, and limit load analysis ([1], [2]). The general model provides a monotone relation between the evolution of plastic deformation and contact and a global measure of system energy. For the optimization problems the variational formulation is usually added as so-called state equation constraints by including a set of necessary conditions for the variational statement, using both stress, displacement, plastic multipliers and contact pressure as state variables (see e.g. Refs. [3] – [6] for sensitivity analysis and design for the case of contact, and Refs. [7] – [9] in the case of elasto-plastic behaviour). The approach advocated in this paper, however, treats the analysis problem in its original variational form. For certain design problems it is demonstrated that a more direct approach can be taken which combines the analysis and design goals into one broader variational statement. For a design problem formulated as maximization of the load carrying capacity of a structure under certain constraints, the resulting optimization problem becomes especially transparent and gives rise to very interesting relations between so-called design constraints and unilateral constraints arising from the plasticity and contact conditions.

It should be stressed that with a view to perform simultaneous analysis and design, a mixed variational model is less attractive because it implies a max-min formulation. Our goal has been to develop a *pure max* or *pure min* formulation such that *state-* and *design quantities* can be treated as *simultaneous variables* in the computational procedure. This is a potential advantage in terms of computational cost relative to, e.g., usual schemes where the equilibrium equations are solved "exactly" for each step of redesign, although, at the intermediate steps of redesign, this is not necessary because the design may be far from the optimal solution. Although the method proposed in this paper involves treatment of all the variables at the same time, the computertime may be reduced due to saving of a large number of iterations required for sequences of "exact" solutions of the equilibrium equations for intermediate designs.

2. VARIATIONAL MODELS.

For the sake of notational simplicity we consider truss structures (FEM discretized continuum structures). For the analysis problem, a mixed variational model in bar forces q and nodal displacements x can be expressed as (cf. [10]):

$$\max_{q} \min_{x} \left\{ x^T Bq - \frac{1}{2} q^T Qq - f^T x \right\} \tag{1}$$

where B denotes the compatibility matrix, Q the compliance matrix (i.e. $\frac{1}{2} q^T Qq$ is the complementary energy) and f the external nodal forces. This mixed formulation can be extended to elasto-plasticity with frictionless contact at nodal points by adding yield constraints of the form

$$\left| q_i / A_i \right| \leq \bar{\sigma} , \quad i = 1, \ldots , NB \tag{2}$$

where $\bar{\sigma}$ is a given yield stress, A_i bar areas, and NB the number of bars, along with conditions

$$x_j \leq \bar{x}_j , \quad i = 1, \ldots , NC \tag{3}$$

for the unilateral contact constraints. In (3), $\bar{x}_j$ are given initial gaps between nodal points and potential contact surfaces, and NC is the number of contact conditions. If no contact surfaces are specified the minimization over x in (1) implies $Bq = f$, and we have the well-known case of holonomic elasto-plastic analysis ([11]):

$$\min_{q} \left\{ \frac{1}{2} q^T Qq \mid Bq = f; \mid q_i / A_i \mid \leq \bar{\sigma} , \quad i = 1, \ldots , NB \right\} . \tag{4}$$

In the analysis of elasto-plastic structures, the limit load problem plays a key role ([11]) and is formulated as

$$\max_{\alpha, q} \left\{ \alpha \mid Bq = \alpha f; \mid q_i / A_i \mid \leq \bar{\sigma} , \quad i = 1, \ldots , NB \right\} . \tag{5}$$

A mixed form of this problem is

$$\max_{q} \min_{x} \left\{ x^T Bq \mid f^T x = 1; \mid q_i / A_i \mid \leq \bar{\sigma} , \quad i = 1, \ldots , NB \right\} . \tag{6}$$

The limit-load problem (5) and the elasto-plasticity problem (4) can be combined into one unified maximization problem (cf. Refs. [1], [2]):

$$\max_{\alpha, q} \left\{ \alpha \mid Bq = \alpha f; \frac{1}{2} q^T Qq \leq \bar{\epsilon}^2; \mid q_i / A_i \mid \leq \bar{\sigma} , \quad i = 1, \ldots , NB \right\} . \tag{7}$$

In this formulation the given constraint value $\bar{\epsilon}^2$ for the complementary energy controls the specific model problem that is handled by the variational statement. For small $\bar{\epsilon}$, (7) is an elasticity problem, and for large $\bar{\epsilon}$, where the constraint $\frac{1}{2} q^T Qq \leq \bar{\epsilon}^2$ is inactive, we have the limit load problem. For intermediate values of

$\bar{\epsilon}$. (7) is an elasto-plasticity model, and we have a monotone relation between the increase in the load carrying capacity α and the global measure $\bar{\epsilon}^2$ of system energy.

If we now include the contact condition, the displacement variable x can be removed from (1) by use of conjugate duality (A. Ben-Tal; private communication). In the case of symmetric bounds on x, i.e. $|x_j| \leq \bar{x}_j$, $j = 1, \ldots, NC$, we get a formulation

$$\max_{\alpha, q} \left\{ \alpha - \Sigma \bar{x}_j \, |(Bq - \alpha f)_j| \ \bigg| \ \frac{1}{2} q^T Qq \leq \epsilon^2; \ |q_i / A_i| \leq \bar{\sigma}, \ i = 1, \ldots, NB \right\} \quad (8)$$

where we set $\bar{x}_j = \infty$ in the case of no contract condition. If the constraint on complementary energy is not active we have defined a limit load problem for plasticity with possible contact.

3. FORMULATION OF UNIFIED ANALYSIS PROBLEM BY MEANS OF CASTIGLIANO'S 2ND THEOREM

The pure minimum and maximum character, respectively, of the variational formulations of the analysis problems (4) and (7), may be preserved even if contact conditions are considered, if we make use of Castigliano's 2nd theorem to express the displacement conditions (3) in terms of forces. A similar advantage is achieved in the formulation (8), but we now consider one-sided bounds (3) on x_j, and entirely base our formulation on principles of mechanics.

Let us denote by r_j the possible external contact forces exhibited by frictionless plane surfaces that may constrain nodal displacements of the truss. If contact occurs, the equations of equilibrium changes to

$$Bq = f + ar \quad (9)$$

where the matrix a projects the contact forces onto the directions of the external nodal forces f. The potential contact forces are orthogonal to the frictionless contact surfaces and taken to be nonnegative.

$$r_j \geq 0 \qquad j = 1, \ldots, NC . \quad (10)$$

when directed along the outward normal. Non-zero contact forces r_j imply a change in the complementary energy $\frac{1}{2} q^T Qq$ through (9), and according to Castigliano's 2nd theorem the nodal point displacements x_j in the directions of the external forces r_j are simply given by

$$x_j = \frac{\partial (\frac{1}{2} q^T Qq)}{\partial r_j} = \left[\frac{\partial q}{\partial r_j} \right]^T Qq , \qquad j = 1, \ldots, NC . \quad (11)$$

Since r_j and hence x_j is directed away from the contact surface, the kinematic contact condition (3), where the initial gap $\bar{x}_j$ is given, is now expressed in terms of forces,

$$- \left[\frac{\partial q}{\partial r_j} \right]^T Qq \leq \bar{x}_j \qquad j = 1, \ldots, NC \; . \tag{12}$$

In addition to (10) and (12), the unilateral contact problem is characterized by the fact that $r_j > 0$ and $r_j = 0$, respectively, depending on whether the condition (12) is satisfied as an equality or an inequality. Thus, as an additional governing condition for the contact problem, we have

$$r_j \left[\left[\frac{\partial q}{\partial r_j} \right]^T Qq + \bar{x}_j \right] = 0 \; , \qquad j = 1, \ldots, NC \; . \tag{13}$$

The variational formulation of the elasto-plastic analysis problem (4) may now be generalized as follows to allow for contact conditions,

$$\min_{q,r} \left\{ \frac{1}{2} q^T Qq \; \middle| \; Bq = f + ar; \; |q_i / A_i| \leq \bar{\sigma}, \; i = 1, \ldots, NB; \; r_j \geq 0 \; . \right.$$

$$\left. - \left[\frac{\partial q}{\partial r_j} \right]^T Qq \leq \bar{x}_j \; , \; r_j \cdot \left[\left[\frac{\partial q}{\partial r_j} \right]^T Qq + \bar{x}_j \right] = 0 \; , \; j = 1, \ldots, NC \right\} \; . \tag{14}$$

Similarly, we may extend the max formulation (7) to the following model that unifies the limit load problem with those of elasto-plasticity and unilateral contact,

$$\max_{a,q,r} \left\{ \alpha \; \middle| \; Bq = \alpha f + ar; \; \frac{1}{2} q^T Qq \leq \bar{\epsilon}^2; \; |q_i / A_i| \leq \bar{\sigma} \; , \; i = 1, \ldots, NB; \right.$$

$$\left. r_j \geq 0 \; , \; - \left[\frac{\partial q}{\partial r_j} \right]^T Qq \leq \bar{x}_j \; , \; r_j \cdot \left[\left[\frac{\partial q}{\partial r_j} \right]^T Qq + \bar{x}_j \right] = 0 \; , \; j = 1, \ldots, N \right\} \tag{15}$$

In the final statement of (14) and (15), the terms $\partial q / \partial r$ should be eliminated by suitable rearrangement of the equilibrium condition and separation of forces in non-redundant and redundant forces (cf. [13]).

To illustrate a solution to the analysis problem (15), consider the plane five-bar truss in Fig. 1 with specific elastic bar stiffnesses EA, lengths $\ell_i = (1, \sqrt{2}, 1, \sqrt{2}, 1)\ell$, and large yield stress $\bar{\sigma}$ such that elasticity prevails. The truss is subjected to a given vertical external load f_1 at the upper, right hand nodal point, for which a vertical frictionless contact surface is defined along with the initial gap $\bar{x}$. The essence of the solution to the elastic contact problem is depicted in the figure through the dependence on the load parameter of the bar forces q_i, $i = 1, \ldots, 5$, and the horizontal displacement h and vertical displacement v of the upper right hand nodal point.
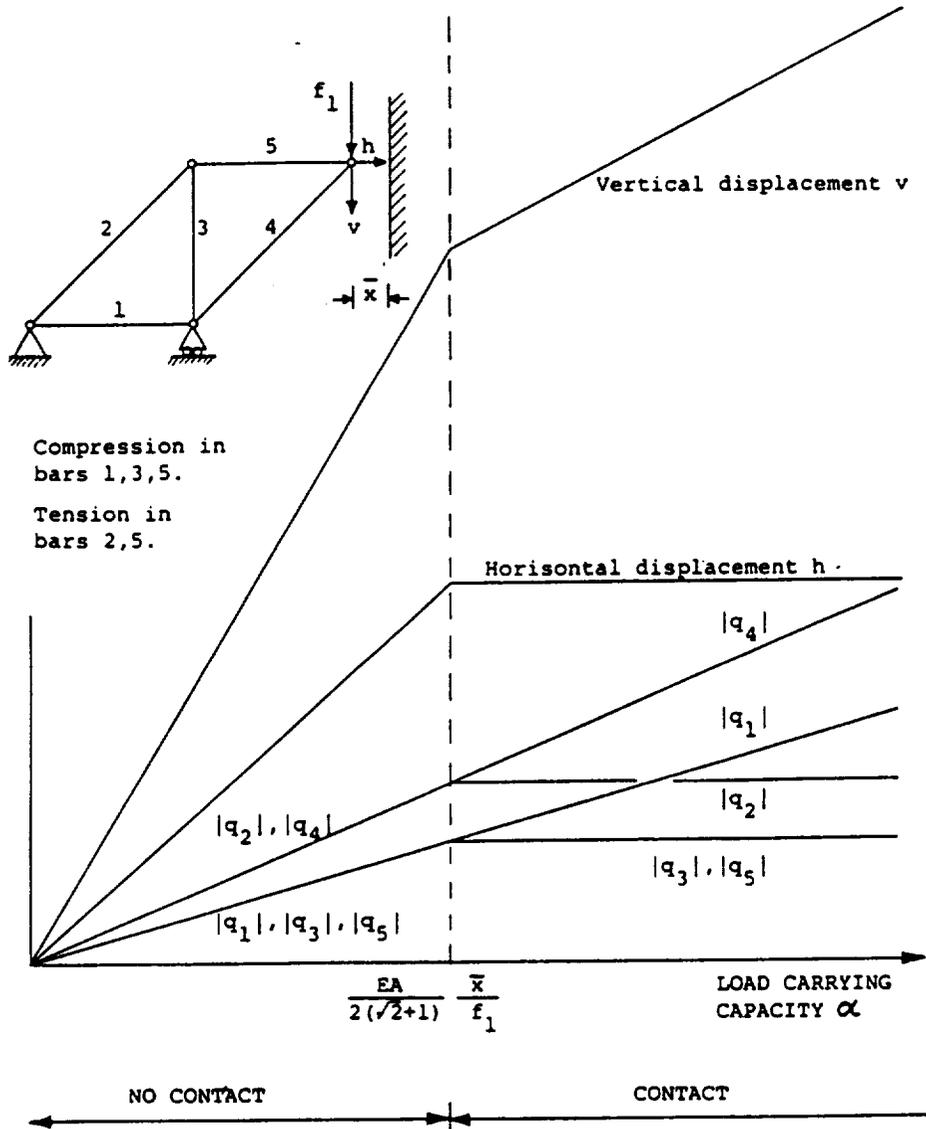
fig.1.

4. DESIGN FORMULATION.

The variational statements of maximization of the load carrying capacity constitutes a natural basis for the formulation of a unified analysis and design problem (see Ref. [12], Chapt. 10 for a discussion on the advantages of such a setting). Denoting by A_i the cross-sectional areas of the bars in a truss, a combined problem formulation, with A_i, $i = 1, \ldots , NB$, as design variables, takes the form:

344

$$\text{maximize} \quad \alpha$$
$$\alpha, q, r, A_i$$

$$\text{so}: \quad \left\{ \begin{array}{l} \text{Constraints of variational} \\ \text{problem} \end{array} \right\}$$

$$\left\{ \begin{array}{l} \text{Constraints on design:} \\ \text{NB} \\ \sum_{i=1}^{NB} l_i \, A_i = \bar{V} \, , \quad \underline{A} \leq A_i \leq \bar{A} \end{array} \right\} \quad .$$

This problem maximizes the load carrying capacity of a given structure, for a giveen compliance and for a given volume. For a fixed volume, the compliance constraint controls whether the structure is in the elastic, elasto-plastic or limit load range, with or without contact, thus encompassing a broad class of model problems. Note that the problem includes displacement as well as stress constraints, but as *part of the analysis model*. Unlike in traditional design formulations an active displacement constraint gives rise to a contact force and an active stress constraint gives rise to a plastic deformation. *The problem thus finds the ultimate load carrying capacity, allowing the structure to yield and to explore the possibility of advantageous contact forces.* If displacement and stress constraints are to be included as part of the design model, minor modifications need to be included in the problem statement that has the effect of nullifying contact forces and plastic deformation from such constraints (cf. [13]).

Fig. 2 shows results for a case where contact is not present and illustrates the dependence of the optimal load carrying capacity α on the compliance constraint value $\bar{\epsilon}$ and the volume constraint value $\bar{V}$, respectively. Note that for an increase in volume the structure moves from the limit load range through an elasto-plastic range to the elastic range and through yet another elasto-plastic range before settling in the elastic range. The intermediate step is caused by one of the design constraints $A_i \leq \bar{A}$ becoming active.
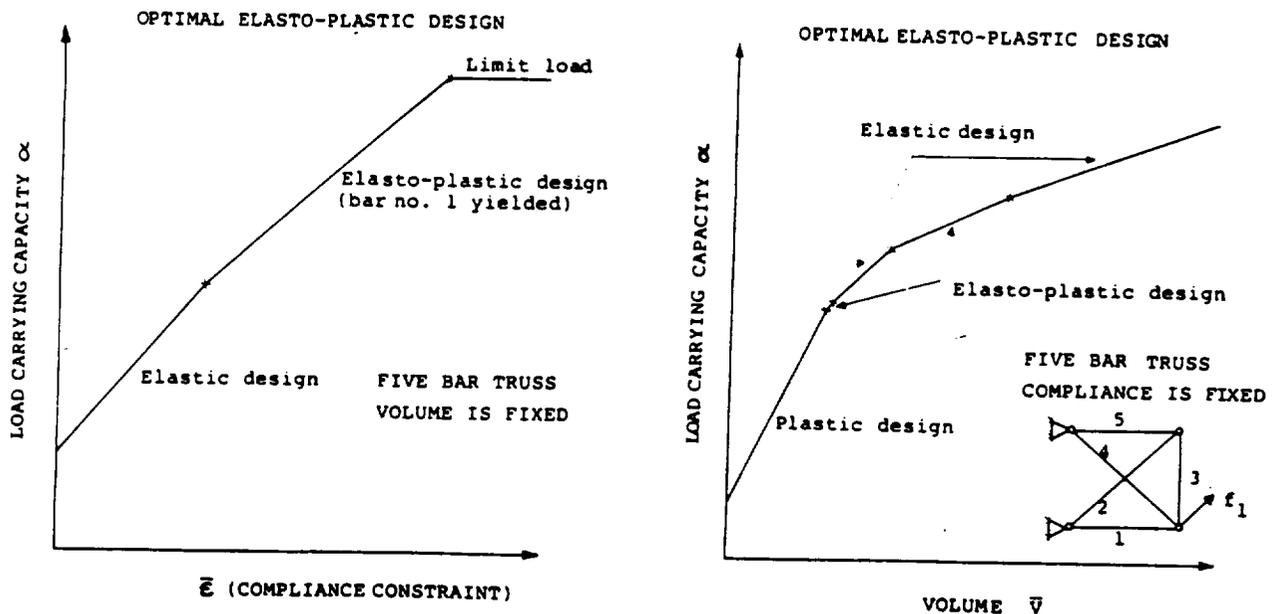
fig.2.

REFERENCES

[1] J.E. Taylor: Towards a Unified Model for Elasto-Plasticity Structural Analysis. Mechanics Research Communications, 16 (2), 1989, pp. 125-131.

[2] A. Ben-Tal, J.E. Taylor: Towards a Unified Model for the Elasto-Plastic Analysis of Structures. Report. Technion, Israel, 1989.

[3] M.P. Bendsøe, N. Olhoff, J. Sokolowski: Sensitivity Analysis of Problems of Elasticity with Unilateral Constraints. Journal of Structural Mechanics, 13 (1985), pp. 201-222.

[4] R.L. Benedict, J.E. Taylor: Optimal Design for Elastic Bodies in Contact. Proc. of NATO-ASI, Iowa City, Iowa, 1980. Optimization of Distributed Parameter Structures - Vol. 2, (Eds. E.J. Haug, J. Cea), Sijthoff and Noordhoff, Netherlands, pp. 1553-1589, 1981.

[5] P. Neittaanmaki: Optimal Shape Design in Contact Problems. In: Discretization Methods and Structural Optimization - Procedures and Applications, (eds. H.A. Eschenauer, E. Thierauf), Lecture Notes in Engineering, Vol. 42, Springer Verlag, 1988, pp. 248-254.

[6] Hlavacek, Bock, Lovisek: Optimal Control of Variational Inequality with Applications to Structural Analysis. Part 1, Part 2, Part 3, Applied Mathematics and Optimization, Vol. 11, 1984, pp. 111-143, and Vol. 13, 1985, pp. 117-136.

[7] M.P. Bendsøe, J. Sokolowski: Design Sensitivity Analysis of Elastic Plastic Analysis Problems. Mechanics of Structures and Machines, 16 (1988), pp. 81-102.

[8] I. Kaneko and G. Maier: Optimum design of plastic structures under displacement constraints, Comput. Methods Appl. Mech. Eng. 27, pp. 369-391 (1981).

[9] C. Cinquini, M. Rovati: Optimal Design of Elastic Plastic Structures. In: Computer Aided Optimal Structural Design, (eds. N. Olhoff, S. Kibsgaard), Lecture Notes, European Program for Advanced Education, Aalborg University, 1989.

[10] K. Washizu: Variational Methods in Elasticity and Plasticity. Pergamon, Oxford, 1983.

[11] M.Z. Cohn and G. Maier (Eds.): Engineering Plasticity by Mathematical Programming. Proceedings of the NATO Advanced Study Institute, University of Waterloo, Ontario, Canada, 1977. Pergamon Press, Oxford, England, 1979.

[12] R.T. Haftka, Z. Gürdal, M.P. Kamat: Elements of Structural Optimization. 2nd Edition, Kluwer, Dordrecht, The Netherlands, 1990.

[13] M.P. Bendsøe, N. Olhoff, J.E. Taylor: A Method for Simultaneous Analysis and Design of Elasto-Plastic Structures with Contact. (In preparation.)

Improving Stability and Strength Characteristics of Framed Structures with Nonlinear Behavior

by

Shahram Pezeshk

Department of Civil Engineering
Memphis State University, Memphis, TN 38152

INTRODUCTION

In this paper an optimal design procedure is introduced to improve the overall performance of nonlinear framed structures. The design methodology presented here is a multiple-objective optimization procedure whose objective functions involve the buckling eigenvalues and eigenvectors of the structure. A constant volume with bounds on the design variables is used in conjunction with an optimality criterion approach. The method provides a general tool for solving complex design problems and generally leads to structures with better limit strength and stability.

Many algorithms have been developed to improve the limit strength of structures. In most applications geometrically linear analysis is employed with the consequence that overall strength of the design is overestimated. Directly optimizing the limit load of the structure would require a full nonlinear analysis at each iteration which would be prohibitively expensive. The objective of this paper is to develop an algorithm that can improve the limit-load of geometrically nonlinear framed structures while avoiding the nonlinear analysis.

One of the novelties of the new design methodology is its ability to efficiently model and design structures under multiple loading conditions. These loading conditions can be different factored loads or any kind of loads that can be applied to the structure simultaneously or independently.

Attention is focused on optimal design of space framed structures. Three-dimensional design problems are more complicated to carry out, but they yield insight into real behavior of the structure and can help avoiding some of the problems that might appear in planar design procedure such as the need for out-of-plane buckling constraint. Although researchers in the field of structural engineering generally agree that optimum design of three-dimension building frames especially in the seismic regions would be beneficial, methods have been slow to emerge. Most of the research in this area has dealt with the optimization of truss and plane frame structures.

FORMULATION AND DEVELOPMENT

To improve the limit load behavior and stability characteristics of a 3-D framed structure under multiple loading conditions, we need to consider several ingredients to generate an objective function. Based on observations made on the limit and post-limit behavior of elasto-plastic frames, Hjelmstad and Pezeshk (1988) developed an approximate model of nonlinear behavior of framed structures. From the model they observed that the overall stability and strength of a structure can be improved by maximizing the linear buckling eigenvalues of the structure. This concept was applied by Pezeshk and Hjelmstad (1989) to improve the performance of planar framed structures which worked successfully. Thus, in order to improve the limit-load and post-limit behavior of framed structures we must include the buckling eigenvalues in the objective function. To handle multiple loading cases, we also purpose including the loading conditions directly in the objective function. We further hypothesize that the loading conditions should become important to the objective function if they cause displacements similar to a buckling eigenvector which is in the design subspace. The proposed formulation then, seeks to maximize buckling eigenvalues of the structure using the work of the various load cases going through modal displacements as weighing factors.

Natural way to consider all the ingredients discussed above in the framework of an objective function would be to combine them in a quadratic form. The only shortcoming of using a quadratic form is that the number of the buckling eigenpairs used in the objective function is not necessarily equal to the number of the loading conditions. Therefore, it is best to consider a quasi-quadratic form such as

$$\sum_i^N \sum_j^L \mu_i \, \beta_{ij} \, \mu_{\pi_{i_j}} \qquad where \qquad \beta_{ij} = |f_j \cdot \phi_i| \tag{1}$$

where N is the number eigenmodes considered, L is the number of loading conditions, and π_j is a vector containing a set of index numbers with values in the range [1,N] with one-to-one correspondence with the buckling eigenvalues. The dimension of the set is equal to the number of loading conditions (L). For example

$$\text{II-2,3} \quad \textit{leads to} \quad \rightarrow \quad \sum_{i=1}^{N} \mu_i \left[(f_1.\phi_i)\mu_2 + (f_2.\phi_i)\mu_3 \right] \tag{2}$$

Therefore, in order to improve the stability and strength characteristics of a framed structure we define the following optimization problem:

$$\textit{Maximize} \qquad \sum_{i=1}^{N} \sum_{j=1}^{L} \mu_i(x)\beta_{ij}(x)\mu_{\pi_j}(x) \tag{3}$$

$$\textit{Such That} \qquad \sum_{i=1}^{M} A_i(x_i)\Omega_i - \Gamma \tag{4}$$

$$\underline{x}_{ij} < x_{ij} < \bar{x}_{ij} \tag{5}$$

where μ_i=ith elastic critical buckling eigenvalue; Γ=given weight of the structure; A_i=area of group i.

In this paper the members of the structures are assembled into M distinct groups. Each group is associated with a set of design variables which describe the geometry of the cross section of that group. For example an I-beam can be described by its depth h, flange width b, web thickness t, flange thickness t_f. Consequently, the I-beam has four design variables (v=4, where v is the total number of independent design variables). A rectangular cross section has two design variables (v=2): the width b and the height h of the cross section. The vector of design variables will be designated as $x = \{x_1, x_2, ..., x_M\}$ and x_i is the design vector for group i where $x_i = \{x_{i1}, x_{i2}, ..., x_{iv}\}$. x_{ij} is the jth design variable for ith group. To simplify notation, we designate the *specific weight* of the mth group as the weight per unit of cross sectional area of the entire group

$$w_m - \sum_{i \in m} \rho_i L_i \tag{6}$$

where the length of member i is L_i, and its density is ρ_i. The sum is taken over all members associated with group m.

The optimization problem considered here is atypical of multi-objective optimization problems because all of the objective functions have the same nature and yet are conflicting. For example, maximizing one buckling eigenvalue might result in a decrease in another one. A good survey of different generating techniques can be found in Atrek, *et al.* (1984) and Cohon (1975). Since all the objective functions are of the same nature, a weighing technique is best method to generate the *noninferior set* or *Pareto optimal set*. One of the advantages of the formulation developed here is that all the weighing factors are determined automatically, eliminating the principal difficulty inherent in a general weighing solution technique.

Using Eq. (3) and Eq. (4) the Lagrangian functional can be cast as

$$L(x,\xi) - \sum_{i=1}^{N} \sum_{j=1}^{L} \mu_i(x)\beta_{ij}\mu_{\pi_j}(x) - \xi \left[\sum_{i=1}^{M} A_i(x_i)\Omega_i - \Gamma \right] \tag{7}$$

where ξ is the Lagrange multiplier. It should be pointed that the constraints on the size of elements given in Eq. (5) are not included in deriving Eq. (7). Constraints on permissible sizes can be handled efficiently by treating them as passive constraints in the sense that whenever an element violates the size constraints, the design variable associated with that element adopts the minimum or the maximum permissible sizes and is placed in the passive set. Allwood and Chung (1984) have suggested that if a design variable is moved to the passive set in two consecutive iterations, it will probably will be passive at optimum. In principle, the method suggested by Allwood and Chung was followed in the computation reported in this paper. However, it was found that in the early stages of optimization it is best to keep all the design variables as active and follow Allwood and Chung procedure after few optimization cycles when the algorithm settles down.

Differentiating Eq. (7) with respect to design variable x_{mn} and setting the corresponding equation equal to zero results in the optimality criteria which for simplicity of notation can be represented as

$$\frac{Q_{mn}}{\xi} - 1 \qquad m = 1, \, ..., \, M \quad and \quad n = 1, \, ..., \, \nu \tag{8}$$

In Eq. (8) there exit two sensitivity terms: sensitivity of β_{ij} and μ_i with respect to the design variables. The sensitivity of β_{ij} is a function of sensitivity of eigenvectors. Thus, sensitivity of both eigenvalues and eigenvectors are needed. Determination of these sensitivities are discussed in the following section.

EIGENVALUE AND EIGENVECTOR SENSITIVITY ANALYSIS

Evaluation of the optimality conditions require knowledge of the sensitivity, or rate of change, of the buckling eigenvalues and eigenvectors with respect to the design variables. Procedures for computing these sensitivities have been known for some time, but efficient methods of computation continue to be of interest to researchers. A complete and detailed discussion of the problem has been given recently by Dailey (1989). Consider the following eigenvalue problem

$$K\phi - \mu G\phi \tag{9}$$

where K is the (positive definite) elastic stiffness matrix, and G is the (possibly indefinite) geometric stiffness matrix. Both of matrices are symmetric and depend on design variables x. By differentiating Eq. (10) with respect to the design variables and some mathematical manipulations we can determine the eigenvalue and eigenvector sensitivities as

$$\mu_i' - \frac{\phi_i'[K' - \mu_i G']\phi_i}{\phi_i' G \phi_i} \qquad and \qquad \phi_i' - \sum_{j \neq i} \frac{\phi_j'[K' - \mu_i G']\phi_i}{(\mu_j - \mu_i)\phi_j' G \phi_j}\phi_j - \frac{1}{2}\frac{\phi_i' G' \phi_i}{\phi_i' G \phi_i}\phi_i \tag{10}$$

where a prime indicates differentiation with respect to the design parameter.

RECURRENCE RELATIONS

The optimality criteria are used to modify the design variables in each direction in terms of recurrence relations similar to that proposed by Khot (1981):

$$x_{mn}^{\kappa+1} - x_{mn}^{\kappa}\left[1 + \frac{1}{r}(\frac{Q_{mn}}{\xi} - 1)\right] \tag{11}$$

where κ denotes the iteration number and r is the step size parameter. The convergence behavior depends on parameter r. Depending on the behavior of the constraint, it may be necessary to increase r in order to prevent convergence. If the optimization problem is run in a non-iterative environment a large value of step length such as $r=8$ or $r=10$ is recommended to ensure that there is no problem with divergence. Of course, this choice will result in slower convergence.

EQUATION TO DETERMINE LAGRANGE MULTIPLIER

In order to be able to use the recurrence Eq. (11), the Lagrange multiplier ξ has to be determined. The Lagrange multiplier is determined by using the condition that after each iteration the design moves on the constraint surface so that the constraint is satisfied. A set of equations to determine the Lagrange multiplier can be obtained by linearizing the constraint equation about current iterate. Linearizing the volume constraint about the configuration x^{κ} one obtains

$$L[C]_{x=x^{\kappa}} - C(x^{\kappa}) + \sum_{m=1}^{M}\sum_{n=1}^{\nu}\frac{\partial C}{\partial x_{mn}}(x_{mn} - x_{mn}^{\kappa}) \tag{12}$$

349

The Lagrange multiplier can be obtained by satisfying the linearized constraint equation at the new iterate $x^{\kappa+1}$. Substituting for the appropriate terms and simplifying, one gets

$$\xi = \frac{\sum\limits_{m=1}^{M} \sum\limits_{n=1}^{\nu} \frac{\partial A_m}{\partial x_{mn}} x_{mn} Q_{mn} \Omega_m}{\sum\limits_{m=1}^{M} \left[r A_m + \sum\limits_{n=1}^{\nu} \frac{\partial A_m}{\partial x_{mn}} x_{mn} \right] \Omega_m - r \Gamma} \tag{13}$$

since the constant volume constraint is an equality, ξ can be either positive or negative.

SCALING PROCEDURE

After each iteration to satisfy the constraint relationship, it is necessary to scale design variables to bring the volume of the structure to the level of the assigned volume constraint Γ. Scaling is necessary to insure that the design at each iteration is feasible. The following is a development of the scaling procedure for rectangular members. The same procedure can be developed for I-beam cross sections.

The weight of the structure after each iteration can be divided to three groups depending on which design variables are passive and which are active. Thus, the total weight is given by

$$\Gamma = W^{aa} + W^{ap} + W^{pp} \tag{14}$$

where W's are various weights. A superscript "a" indicates an active design variable, superscript "p" indicated a passive design variable, and there is one superscript for each design variable in the group. The weight of the structure is scaled after each iteration by scaling only the active design variables. The scaling factor η, such that $x_{ij} \leftarrow x_{ij}\eta$, is determined by the equation:

$$\eta = \sqrt{\frac{\Gamma - W^{pp}}{W^{aa}} + \left[\frac{W^{ap}}{2W^{aa}} \right]^2} \frac{W^{ap}}{2W^{aa}} \tag{15}$$

The scaling equation for n design variables per group is an nth order polynomial. Higher order polynomials can be easily solved by Newton's method.

In the following sections the optimization procedure is applied to an irregular framed structure. The purpose of the example problem is to demonstrate the performance of the optimization procedure.

SETBACK FRAME EXAMPLE

It is often difficult to identify the design changes necessary to improve the performance of a structure, especially when the structure is irregular and the response is nonlinear. The frame considered here is a two-story setback frame as shown in Fig. 1. The topology of the frame was picked from a report by Cheng and Truman (1985) and redesigned to meet ATC-03-06 earthquake design recommendation (1978).

A preliminary design was performed using full dead and live load in all members, using approximate coefficients to determine maximum moments in girder sections. Because setback is an irregular structure, a modal analysis procedure was employed to determine earthquake loads. The following seismic coefficients in accordance with ATC-3-06 were used: effective peak acceleration (A_a=4); effective peak velocity-related acceleration (A_v=0.4); soil profile characteristics of site (S_2=1.2); reduction factor to account for effects of inelastic behavior (R=4.5); seismic Category C; and seismicity index of 4. The loads on the structure were: dead load: 80psf, and live load: 40psf. A set of eight combination of load effects, as recommended by ATC-3-06, was considered. The critical load effect due to the application of seismic forces on the building are determined as a combination of prescribed loads: 100% of the force for one direction plus 30% of the force for the perpendicular direction. The eight different loads were applied to the building and the stresses and the displacements of each load combination were determined. Members of the building were checked for the worst loading case and were redesigned if necessary. This procedure of analysis and redesign was carried out for several iterations until all the requirements were satisfied. The member properties of the final design

were checked against the AISC (1978) specification and all the requirements were satisfied. The properties of the design are given in Table 1. Load cases I and II are shown in Fig. 2.

Single Loading. The setback frame was optimized under single loading condition I. The properties of optimized design is summarized in Table 1. Both Initial design and optimized designs were analyzed and the their nonlinear performances are given in Fig. 3. From this figure one can observe that optimized design performed better than the initial design with better load carrying capacity and improved post-limit behavior.

Multiple Loading. Most optimization procedures developed in recent years are set to minimize the volume of a structure under some displacement and stress constraints. Displacement constraint optimization methods have been used to limit displacement of a design to minimize damage or perhaps to force a design to remain in the elastic range. Unfortunately, such an approach does not assure overall structural stability. Under severe multiple loading conditions, the displacement constrained optimal design may not have desirable global stability characteristics. To demonstrate this point, the initial setback design was optimized with a single displacement constraint optimization procedure to compare its performance with the optimized design based on stability.

The weight of the structure was minimized with a top displacement of 1.2 inches, under a combination of load cases I and II, resulting in an optimized structure of the same weight as those optimized for stability. The properties of the optimized design under displacement constraints along with the properties of the optimized design based on stability are given in Table. 1. The nonlinear performance of both optimized designs are shown in Fig. 4. The structure optimized for stability performed better under both loading condition better than the initial design with better load carrying capacity and about the same post-limit behavior. On the other hand, the displacement constraint design was stiffer and stronger than initial design under load case I but quite poor compared to the initial design under load case II. Furthermore, the displacement constraint design demonstrated a more brittle behavior under load case I than both the initial design and stability design. This example can serve to illustrates the strength of the stability design procedure in improving global stability and strength of a structure under multiple loading cases.

CONCLUSION

An optimization-based design procedure is presented that can effectively and efficiently produce structural designs with better overall strength and stability characteristics than alternative structures of the same weight. The method can improve the stability performance of framed structures under single and multiple loading conditions. The optimization procedure is specially effective for tall buildings when geometry effect is important. The setback frame optimized here is not a tall building, yet the optimization procedure improved the overall performance and the stability characteristics of the building frame.

LIST OF REFERENCES

1. ATC-3-06. (1978). Tentative Provisions for Development of Seismic Regulations for Buildings, Prepared by the Applied Technological Council, NSF, and NBS, Special Publication, NSF publication 78-8.
2. Allwood, R.J., and Y.S. Chung. (1984). "Minimum-Weight Design of Trusses by an Optimality Criteria Method." *International J. for Numerical Methods in Eng.*, 20, pp 697-713.
2. Atrek, E., R.H. Gallagher, K.M. Ragsdell, and O.C. Zeinkiewicz. (1984). *New Directions in Optimum Structural Design.* John Wiley & Sons Ltd., New York.
3. Cheng, F.Y., and K.Z. Truman. (1985). "Optimal Design of 3-D Reinforced Concrete and Steel Buildings Subjected to Static and Seismic Loads Including Code Provisions." *Structural Series 85-20.* Dept. of Civil Eng. Rolla, MO.
4. Dailey, R.L. (1989). "Eigenvector Derivatives with Repeated Eigenvectors." *AIAA J.*, 27(2), pp 486-491.
5. Hjelmstad K.D. and S. Pezeshk. (1988). "Approximate Analysis of Post-limit Response of Frames." *J. of Struct. Eng., ASCE*, 114(2).
6. Koski, J. (1981). "Multi-Criterion Optimization in Structural Design." *Proc. Int'l Symp. on Optimum Struct. Design*, Tuscon, AZ.
6. Khot, N.S. (1981). "Optimality Criteria Methods in Structural Optimization." *Technical Report AFWAL-TR-81-3124.*
7. Pezeshk, S. and K.D. Hjelmstad (1989). "Optimal Design of Framed Structures under Multiple Loading Conditions Based on a Stability Criterion." *Report No. SRS-547*, UILU-ENG-89-2006. Univ. of Illinois, Urbana.
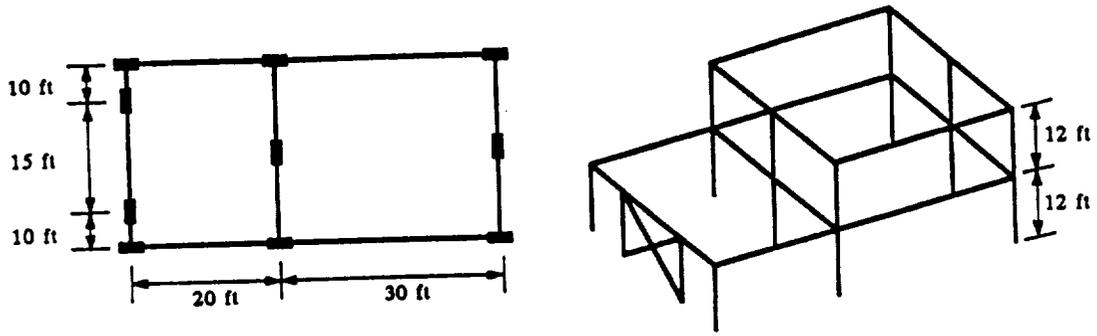
Fig. 1 Topology of the setback frame

Load case I   Load case II
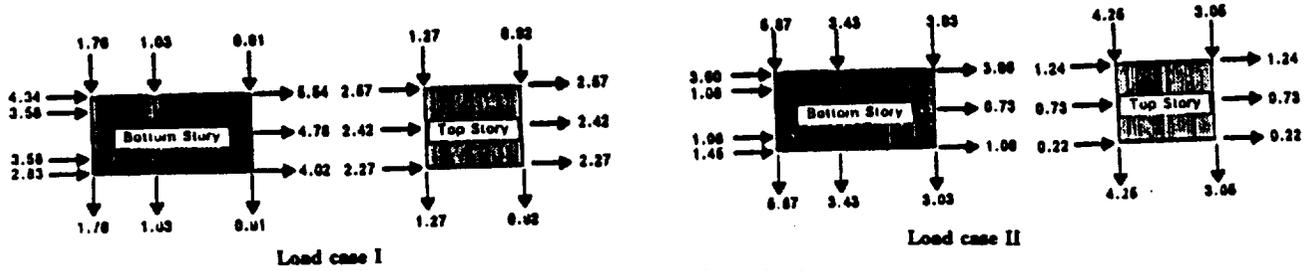
Fig. 2 Load cases for setback structure

Table 1. Properties of the optimized designs for the setback frame.

Properties	Initial Design	Single Load n=10	Multiple Load n=5	Displacement Constraint
1st Story Column				
Width (in)	5.5	6.92	5.62	1.82
Height (in)	9.5	7.50	7.50	19.02
2nd Story Column				
Width (in)	4.0	6.15	4.14	1.90
Height (in)	8.6	7.50	7.50	11.87
1st Story Girders				
Width (in)	1.7	1.90	1.90	1.90
Height (in)	9.8	1.75	7.50	7.50
2nd Story Girders				
Width (in)	1.6	4.11	4.11	5.46
Height (in)	7.5	7.50	7.50	7.50
Braces				
Width (in)	1.6	1.90	1.90	1.90
Height (in)	7.5	7.50	7.50	7.50
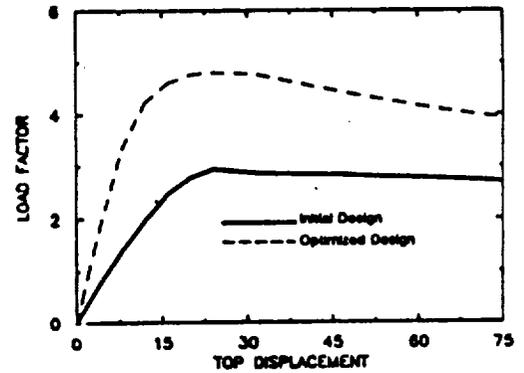
Fig. 3 Response of setback structure under one load case
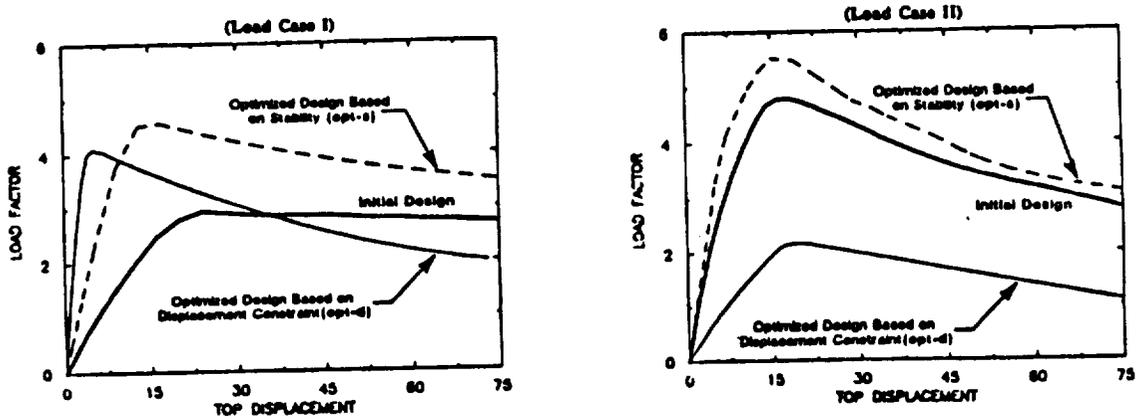
Fig. 4 Comparison of displacement constraint and stability based optimization

352

A STIFFNESS TAILORED WING COVER CONCEPT FOR STRUCTURAL EFFICIENCY AND POSTBUCKLING APPLICATION

D. R. Ambur
NASA Langley Research Center
Mail Stop 190
Hampton, VA 23665
(804) 864-3174

The weight savings due to usage of composite materials in aircraft structural applications is well known. Significant weight and cost benefits are achievable by developing structurally tailored concepts and efficiently integrating them with suitable material and manufacturing technologies. The proposed paper will describe such an efficient concept for application to primary aircraft structures.

The structure selected for analysis is a 3-bay panel from a supersonic aircraft wing shown in Figure 1. This 3-bay panel model is expected to provide a true representation of the central bay behavior and is used to demonstrate buckling resistant and postbuckling designs for tailored wing structures. The dimensions and ply lay-ups of the panel are listed in Table 1 and were obtained from preliminary sizing of this wing structure subjected to several flight load conditions. This preliminary design was constrained by the aeroelastic and strength conditions only. The predominant loading at the location of the panel (circled in figure 1) is inplane compression (N_x) and shear (N_{xy}). Two wing-skin designs were studied - a uniform skin design and a tailored skin design. The uniform skin has a constant thickness 18/36/36/10 (percentage of 0/45/-45/90 plies) ply lay-up. The tailored skin has a 0/45/45/10 ply lay-up with the 0° plies shifted to the spar cap region. Panel weight and prebuckling stiffnesses were constant for the two designs.

The uniform and tailored skin panels were analyzed for buckling and postbuckling responses using the finite element method. The panel designs utilized the T800/5245C graphite-bismaleimide material system. Applied displacements were used in the analysis to simulate a representative loading of N_x=1000 lbs/in. and N_{xy}=1000 lb./in. The three combinations of loading used were N_x (N_x=1000 lbs/in. and N_{xy}=0), N_x+0.5N_{xy} (N_x=1000 lbs/in. and N_{xy}=500 lbs/in.), and N_x+1.0 N_{xy} (N_x=1000 lbs/in. and N_{xy}=1000lbs/in.). The buckling results summarized in Table 2 show that substantial gains are possible even for the N_x+0.5 N_{xy} and N_x+1.0 N_{xy} loading conditions. The buckling factor presented is a ratio of the predicted buckling load to the imposed load.

The postbuckled results obtained from the analysis indicate that the tailored skin panel stiffnesses are better retained into the postbuckled regime than the uniform skin panel stiffnesses. For example, for the $N_x+0.5N_{xy}$ loading condition, initial postbuckling axial and shear stiffnesses are reduced by 50 and 20 percent, respectively, compared to the prebuckling stiffnesses for the uniform skin panel. The corresponding reductions for the tailored skin panel are 28 and 10 percent. The postbuckling performance of the tailored skin panel suggests better suitability of this concept for postbuckled applications. Also, for this load case, the tailored skin has 12 percent improvement in postbuckled strength compared to the uniform skin design. This estimate is based on the application of the maximum strain criterion to the skin first ply failure. The predominantly ±45° ply lay-up (soft skin) for the tailored skin incorporates damage tolerant features into this design as an additional benefit. These performance improvements of the tailored skin design indicate that stiffness tailoring of this type may be very effective for optimizing structural designs.
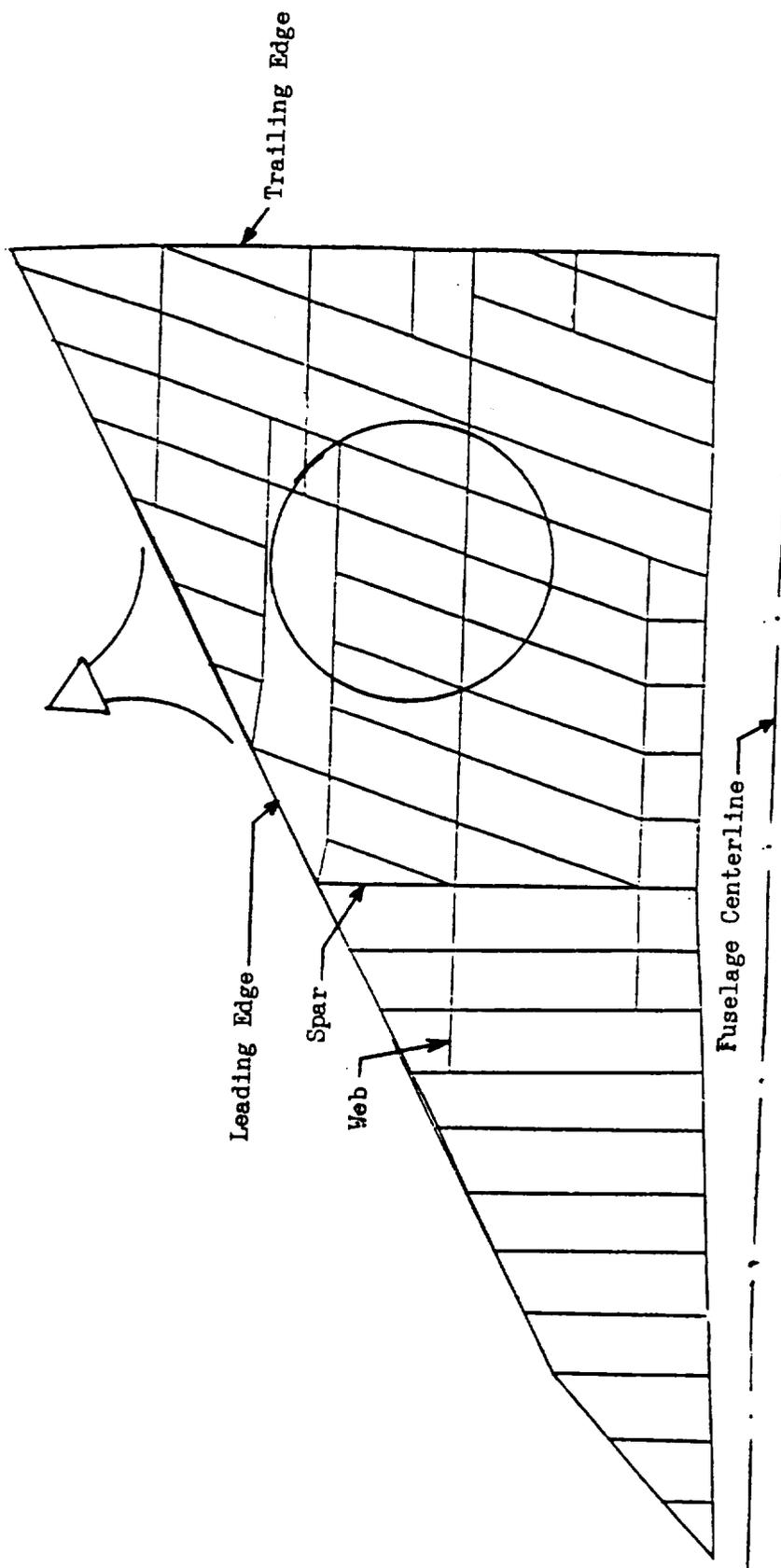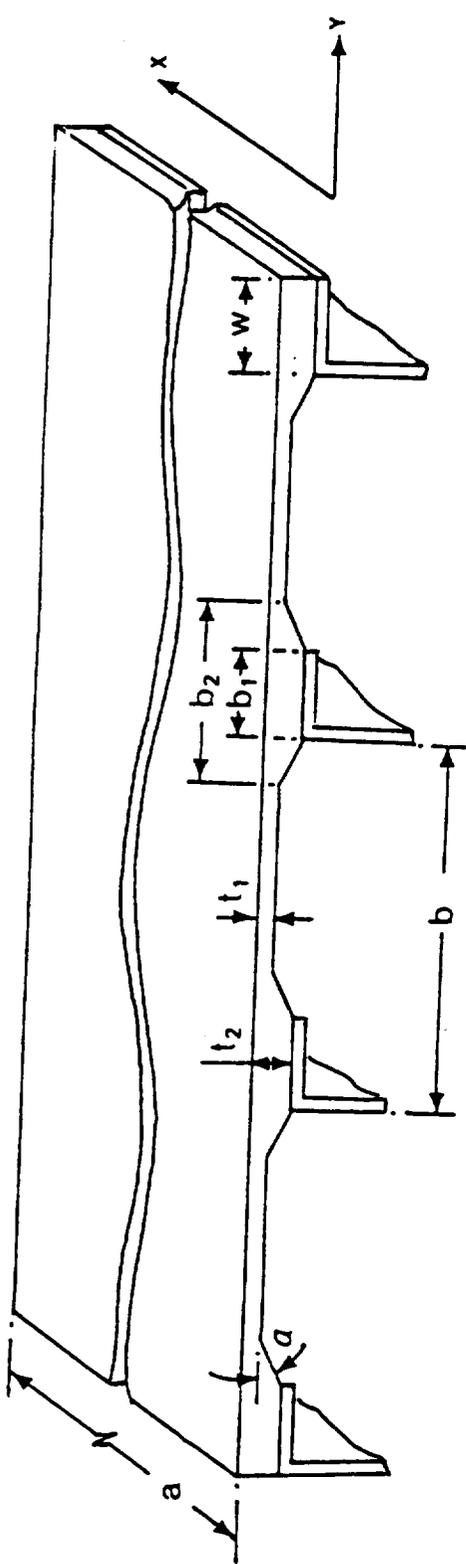
Figure 1. Wing Configuration and Panel Details

Table 1. Geometric and Ply Lay-up Details of Panels

Panel Type	Mid-bay Lay-up, Percentage of 0/45/-45/90 plies	Spar Flange Lay-up, Percentage of 0/45/-45/90 plies	a (in.)	b (in.)	w (in.)	b_1 (in.)	b_2 (in.)	t_1 (in.)	t_2 (in.)	α (deg.)
Uniform Skin	18/36/36/10	18/36/36/10	27.6	9.85	1.18	-	-	0.165	0.165	0
Tailored Skin	0/45/45/10	44/25/25/6	27.6	9.85	1.18	1.18	4.32	0.134	0.24	5

Table 2. Summary of Predicted Panel Buckling Results

Loading[1]	Panel Type	Buckling Factor	Improvement in Buckling Load, percentage
N_x	Uniform	0.310	69
	Tailored	0.525	
$N_x+0.5N_{xy}$	Uniform	0.291	47
	Tailored	0.429	
$N_x+1.0N_{xy}$	Uniform	0.226	17
	Tailored	0.265	

[1] N_x = 1000 lbs/in., N_{xy} = 1000 lbs/in.

Optimal Design of a Space Power System

Young W. Chun
Dept. of Mechanical Engineering
Villanova University
Villanova, PA 19085

James F. Braun
GE Space Systems Division
Valley Forge, PA 19406

INTRODUCTION

The aerospace industry, like many other industries, regularly applies optimization techniques to develop designs which reduce cost, maximize performance, and minimize weight. The desire to minimize weight is of particular importance in space-related products since the costs of launch are directly related to payload weight, and launch vehicle capabilities often limit the allowable weight of a component or system.

With these concerns in mind, this paper presents the optimization of a space-based power generation system for minimum mass. The goal of this work is to demonstrate the use of optimization techniques on a realistic and practical engineering system. The power system shown schematically in Figure 1 uses thermoelectric devices to convert heat into electricity. The heat source for the system is a nuclear reactor. Waste heat is rejected from the system to space by a radiator.

Over the past quarter century, thermoelectrics have been successfully employed to provide electric power to spacecraft on a variety of missions. To date, the primary use of thermoelectrics in space applications has been in radioisotope thermoelectric generators (RTG). RTGs operate by using thermoelectrics to convert the decay heat of a radioisotope fuel such as plutonium-238 into electricity. The Apollo lunar missions employed RTGs, and more recently RTGs were used on the Voyager missions which explored the planets of Jupiter and Saturn. In all of these missions the system electrical power requirements were in the 100W to 1000W range.

As satellites become larger and more sophisticated, the required power levels increase. Future spacecraft in the 1990's are expected to require 100kW to 1MW of electrical power. The use of RTGs to meet these power levels is no longer practical. Alternative power systems such as the SP-100 system are being considered for this task. The aim of the SP-100 project is to design a highly reliable power generation module which will produce approximately 100 kilowatts of electrical power. The power system which is considered in this analysis is based on the

overall characteristics of the SP-100 system.

To perform the minimum mass optimization, a mathematical model of the power generation system is developed which evaluates steady state performance and corresponding system mass. The level of sophistication in the math model is representative of the simplified system definition in the initial design phase.

SYSTEM DESCRIPTION

As indicated in Figure 1, the system is divided into a number of identical segments. Each segment receives a portion of the thermal power output from the reactor and produces a portion of the required electrical power. Realistic system designs have been defined which show 10 and 12 segment configurations. The function of each segment is identical, thus the operation of the total system can be described by the performance of a single segment.

Each segment of the thermoelectric power generation system contains two pumped loops which share a common pump. The two loops are referred to as the primary (hot) and secondary (cold) loops. The primary loop is the high temperature heat supply loop. The secondary loop is the low temperature heat rejection loop. Both primary and secondary loops use a liquid metal as the working fluid. The primary loop working fluid leaves the reactor at its maximum temperature, flows through the pump, into the hot side heat exchanger, and then returns to the reactor at its minimum temperature. As the primary loop fluid flows through the reactor, its temperature is increased as it absorbs the heat output from the reactor. Heat is removed form the primary loop as it flows through the pump and heat exchanger. A portion of this heat is converted into electrical power by the thermoelectric devices. The remaining unconverted portion of this heat is dumped into the secondary loop. The secondary loop working fluid exits the radiator at its minimum temperature and flows into the pump. After exiting the pump, the secondary loop flow is split into two paths through the cold side heat exchangers. Heat is absorbed by the secondary loop (from the primary loop) as it flows through the pump and heat exchangers. The secondary loop is rejoined after flowing through the cold side heat exchangers, and returns to the radiator at its maximum temperature. The secondary loop fluid temperature decreases as it flows through the radiator, since heat is removed from the fluid and rejected to space by radiation.

Thermoelectric devices are sandwiched between the hot and cold heat exchangers. As heat flows through the thermoelectrics a temperature gradient is established across the thermoelectric devices, and these devices convert a portion of this heat into electrical power. Thermoelectric conversion efficiencies

(electrical power output to heat input) typically range from 4% - 8%. An introductory description on the operation and performance of thermoelectrics is provided in [1] and [2].

As the primary and secondary loop fluids flow through the segment, frictional and momentum losses cause pressure drops in each loop. The pump is employed to increase the pressure in each loop, and thus maintain the proper flow rates.

SYSTEM MATHEMATICAL MODEL

To create the system mathematical model, several system components or subsystems are identified, and a separate model is created for each of these subsystems. The subsystems considered in the present work are thermoelectrics, heat exchanger, radiator, pump, reactor, and pipings. The complete system model is formulated by linking all of these subsystem models together through an overall system heat balance. The resulting system model describes the performance and mass characteristics of the entire system.

The total model of the system results in a set of non-linear algebraic equations. Since the equation set is non-linear, the solution of the system is not straightforward and numerical algorithms are required to iteratively determine the system solution.

System Heat Balance. A schematic of a power system segment is shown in Figure 1, including the identification of state points along the primary and secondary loop flow paths. The reactor is the heat source for the system, and it supplies heat to the primary loop of each segment.

Equally dividing the total heat input into each segment, the temperature increase of the primary loop fluid through the reactor can be calculated. This temperature increase is given by

$$Q_{RCT-PLS}/n_{SEG} = m_p c_p [T_{P1} - T_{P3}]$$

(1)

where: $Q_{RCT-PLS}$ = available reactor heat input to the primary loops

n_{SEG} = the total number of power segments in the system
m_p = the primary loop working fluid mass flow rate
c_p = specific heat of primary loop working fluid.

The working fluid temperatures at the indicated state points P1..P3/S1..S3 are designated as $T_{p1}..T_{p3}/T_{S1}..T_{S3}$. The value of T_{p1} is an input, and all other state point temperature values are unknown.

In each segment, the primary loop working fluid flows

directly out of the reactor into the pump. Within the pump, heat is transferred from the primary loop into the secondary loop. A portion of this heat is converted by the thermoelectrics in the pump into electrical power. The difference between the heat lost by the primary loop and the heat gained by the secondary loop is the electrical power produced within the pump. An expression for this power quantity is

$$m_p c_p [T_{P1} - T_{P2}] = P_{E\text{-}PUMP} + m_S c_S [T_{S2} - T_{S1}] \tag{2}$$

where: $P_{E\text{-}PUMP}$ = electrical power produced within the pump
m_s = secondary loop mass flow rate
c_s = specific heat of secondary loop working fluid.

After exiting the pump, the primary loop working fluid enters the hot side heat exchanger (HX). As it flows through the hot HX, heat from the primary loop is transferred from the fluid, through the thermoelectrics, and into the secondary loop fluid. The heat flowing through the thermoelectrics creates a temperature gradient across the thermoelectric devices. The thermoelectrics convert a portion of the primary loop heat input into electrical power. The remaining unconverted portion of heat travels into the secondary loop. Within the HX package of each segment, the difference between the heat removed from the primary loop and the heat gained by the secondary loop is the electrical power generated by a segment. The relationship for this value is given by

$$m_p c_p [T_{P2} - T_{P3}] = P_{E\text{-}TE} + m_S c_S [T_{S3} - T_{S2}] \tag{3}$$

where $P_{E\text{-}TE}$ is the electrical power produced by a single segment.

The quantity of heat from the primary loop that passes into the secondary loop must be rejected from the system as waste heat. The amount of heat rejected per segment, $Q_{REJ\text{-}SEG}$, is determined by

$$Q_{REJ\text{-}SEG} = m_S c_S [T_{S3} - T_{S1}] \tag{4}$$

To complete the system analysis, the temperatures at various junctures and pressure drops throughout the system (especially in heat exchangers, pipings, and radiator) as well as the heat rejection capability of the radiator must be determined. Once the total system pressure drop is determined, then the pump size can be obtained to provide the required power to offset this pressure drop. All of these are done by modelling individual component using appropriate heat transfer equations and pressure drop equations and linking them together. The details of this phase are provided in [1].

The resulting system model has a total of 190 parameters; 83 of these are unknowns and the remaining 107 parameters are system input. In general the input parameters are:

1. Thermal/electrical performance requirements
2. Material properties of system components
3. Dimensional specifications of the system components.

The primary output of the mathematical model is the system total mass. In addition, various temperatures, heat flux, and pressure drops are also calculated.

OPTIMIZATION ANALYSES AND RESULTS

In the optimization phase, 13 parameters were selected as design variables out of 107 system inputs. They are listed in Table 1 and considered to have significant impact on total system mass based on engineering judgments and intuitions. In addition to the upper and lower bounds on these design variables, most of which represent physical constraints due to assembly and manufacturing, two additional constraints are imposed to avoid the invalid system. The first constraint requires that the efficiency of a thermoelectric device within the system does not exceed the maximum theoretical efficiency of a thermoelectric device [3]. The second constraint ensures that the working fluid in the cold secondary loop remains liquid by restricting its temperature to be greater than the melting point.

As a reference point for optimization studies, a baseline design was established by making engineering judgments, which produced a total system mass of 9,515 Kg as shown in Table 2.

Parametric Study. In these studies only the design variable under consideration is varied, while all other design variables remain fixed at their baseline values.

Since these parametric studies do not consider the interdependencies between all of the design variables, they do not produce true optimum designs. However they are still worthwhile endeavors since they can provide some insights to the problem and also the sensitivity of the system mass to variations of the design variables. For example, the system mass is very sensitive to the A_{HX} (HX face area) variation, as shown in Figure 2, but insensitive to changes in the value of A (thermocouple area).

The optimum value of A_{HX} is determined when the increase of the thermoelectric subsystem mass is balanced by the decrease in radiator and pump masses. From Figure 2 it is seen that the mass penalty associated with decreasing the heat exchanger area is more severe than the penalty related to increasing the area.

<u>Multivariable Optimization</u>. A multivariable optimization program based on Generalized Reduced Gradient Method was used to optimize the system producing a design having total mass of 4,058 Kg. The design variable values and component mass are shown in Tables 2 and 3 respectively. It is noted that most of the mass reduction occured in the pump and the thermoelectric power conversion system while those of primary and secondary pipings were slightly increased.

CONCLUSIONS

A mathematical model was created to simulate a space power system. With 13 variables chosen as design variables, traditional parametric study and multivariable optimization were carried out. The results demonstrate the effectiveness of optimization techniques on a realistic and practical energy system. The parametric studies also provide valuable insights to the problem and the sensitivities of the system to the variations of design parameters. Further refinement of the mathematical model combined with increased number of design variables is suggested as future study.

REFERENCES

[1] Braun, J.F., Optimization of a Thermoelectric Power System, MME Thesis, Villanova University, 1988.
[2] Kamal, J., Thermoelectric Energy Conversion, General Electric Co., 1984.
[3] Frass, A.P., Engineering Evaluation of Energy Systems, McGraw Hill Book Co., 1982.

Figure 1
Thermoelectric Power System Segment
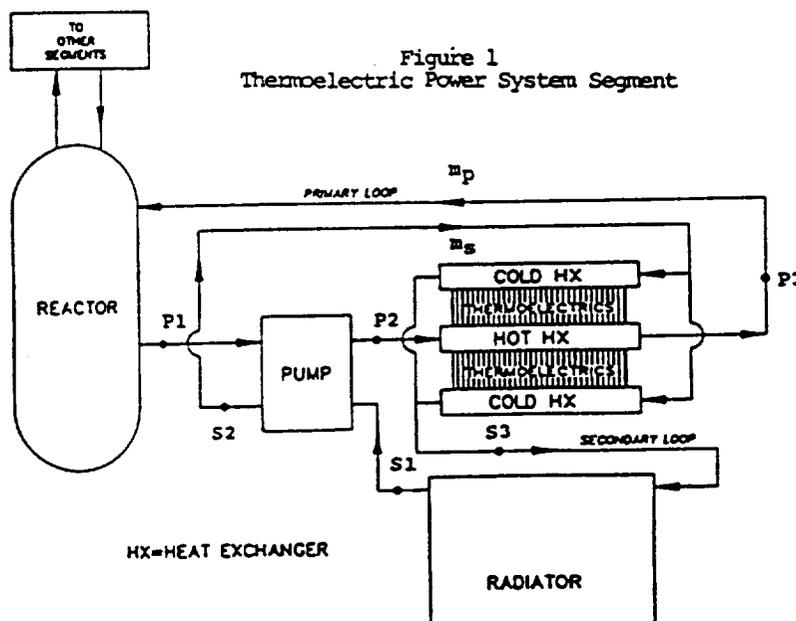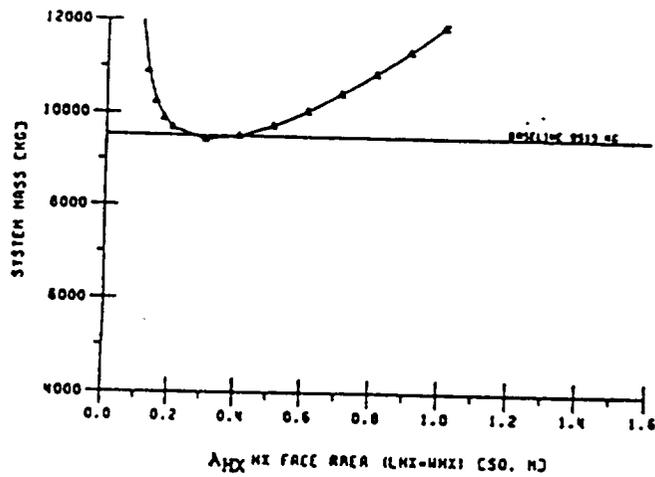
Figure 2
HX Face Area vs. System Mass
(Parametric Study)

Table 2

Multivariable Optimization Results

Variable		Baseline Value	Optimum Value
A_{HX}	m^2	0.4	0.253
d_p	cm	2.0	3.65
d_s	cm	1.5	3.25
d_{TC}	cm	0.6	1.5
l_E	m	0.2	0.424
l_c	m	4.0	5.0
m_s	kg/s	0.79	0.648
m		1.0	1.26
A	mm^2	18.75	19.7
v_H	cm	0.2	0.374
v_c	cm	0.15	0.227
v_R	cm	9.3	0.404
β		1.0	0.88
Total System Mass		9515 Kg	4058 Kg

Table 1

Selected Design Variables for Optimization

Variable	Description	Bounds Lower	Bounds Upper	Units
A_{HX}	HX face area	0.1	1.0	m^2
d_p	Primary loop pipe diam.	0.5	6.0	cm
d_s	Secondary loop pipe diam.	0.5	6.0	cm
d_{TC}	Depth of thermocouple	0.5	1.5	cm
l_E	Radiator evaporator length	0.1	0.5	m
l_c	Radiator condenser length	2.0	5.0	m
m_s	Secondary loop mass flow rate	$.8m_p$	$1.2m_p$	kg/s
m	Load resistance ratio	0.5	2.0	.
A	Area of a single thermocouple	10.0	30.0	mm^2
v_H	Hot HX passage height	0.1	5.0	cm
v_c	Cold HX passage height	0.1	5.0	cm
v_R	Radiator duct passage height	0.1	5.0	cm
β	Thermocouple leg area ratio	0.5	2.0	.

Table 3

Component Mass Per Segment

Component	Baseline Case	Optimum Design
Thermoelectric Power Conversion Subsystem	62.7	29.5
Hot Heat Exchanger	6.9	4.7
Cold Heat Exchanger	13.4	8.7
Radiator	138.6	111.7
Pump	432.8	27.0
Primary Piping System	8.6	15.7
Secondary Piping System	9.9	20.9
Total Mass per Segment	672.9 Kg	218.2 Kg
Mass of 12 Segments	8075 Kg	2618 Kg
Mass of Reactor	1440 Kg	1440 Kg
Total System Mass	9515 Kg	4058 Kg

TOPOLOGY AND BOUNDARY SHAPE OPTIMIZATION
AS AN INTEGRATED DESIGN TOOL

Martin Philip Bendsøe
Mathematical Institute
The Technical University of Denmark
Building 303
DK–2800 Lyngby, Denmark

Helder Carrico Rodrigues
Mechanical Engineering Department
Technical University of Lisbon
Instituto Superior Tecnico
1096 Lisbon Codex, Portugal

Abstract: The optimal topology of a two dimensional linear elastic body can be computed by regarding the body as a domain of the plane with a high density of material. Such an optimal topology can then be used as the basis for a shape optimization method that computes the optimal form of the boundary curves of the body. This results in an efficient and reliable design tool, which can be implemented via common FEM mesh generator and CAD type input–output facilities.

1. Introduction

Traditionally, in shape design of mechanical bodies, a shape is defined by the orientated boundary curves of the body and in shape optimization the optimal form of these boundary curves is computed. This approach is very well established (cf. review paper by Haftka, [1]) and commercial software using this method is available. The boundary variations methods predicts the optimal form of boundaries of a fixed, a priori chosen topology. However, it is well known that the topology is a very important element of the final performance of a mechanical body. As an alternative to the boundary parametrization of shape, a mechanical body can be considered as a domain in space with a high density of material, that is, the body is described by a global density function that assigns material to points that are part of the body. By introducing composites with microvoids, such shape design problems appear as sizing problems for fixed reference domains, and a prediction of topology and boundary shape is possible ([2]–[6]).

The material density approach should be seen as a preprocessor for boundary optimization and by integrating the two methods a very efficient design tool can be developed. In an integrated system, common CAD–style input–output facilities can be used as well as a common mesh generator for the FEM analysis. Interfacing the two methods by a CAD like (based) module added to the input facility for the boundary variations method, allows the designer to actively control the information used and such interactive possibilities have been found to be very important.

2. Topology optimization

For the topology optimization we minimize compliance for a fixed, given volume of material, and use a density of material as the design variable. The density of material and the effective material properties related to the density is controlled via geometric variables which govern the material with microstructure that is constructed in order to relate correctly material density with effective material property.

The problem is thus formulated as

$$\begin{aligned} &\min \quad L(w) \\ \text{so:} \quad &a_D(w, v) = L(v) \quad \text{for all} \quad v \in H \\ &\text{Volume} \le V \end{aligned} \tag{1}$$

where

$$L(v) = \int_\Omega f \, v \, d\Omega + \int_\Gamma t \, v \, d\Gamma \tag{2}$$

$$a_D(w, v) = \int_\Omega E_{ijkl}(D) \, \epsilon_{ij}(w) \, \epsilon_{kl}(v) d\Omega \, . \tag{3}$$

Here, f, t are the body load and surface traction, respectively, and ϵ_{ij} denotes linearized strains. H is the set of kinematically admissable deformations. The problem is defined on a fixed reference domain Ω and the rigidity E_{ijkl} depend on the design variables used. For a so-called second rank layering constructed as in Fig. 1, we have a relation

$$E_{ijkl} \equiv E_{ijkl}(\mu, \gamma, \theta) \tag{4}$$

where μ, γ denote the densities of the layering and θ is the rotation angle of the layering. The relation (4) can be computed analytically ([3]) and for the volume we have

$$\text{Volume} = \int_\Omega (\mu + \gamma - \mu\gamma) d\Omega \tag{5}$$

The optimization problem can now be solved either by optimality criteria methods ([3]) or by duality methods, where advantages is taken of the fact that the problem has just one constraint. The angle θ of layer rotation is controlled via the results on optimal rotation of orthotropic materials as presented in Ref. [7].

It turns out that this method allows for the prediction of the shape of the body and it is possible to predict placement and shape of holes in the structure.

3. Integration

In order to finalize a design obtained by the material density approach, it is reuired to optimize the final shape of the boundaries of the optimal topology. The choice of initial proposed form for the boundary optimization methods is usually left entirely to the designer but the material distribution optimization gives the designer a rational basis for the choice of initial form.

Interfacing the topology optimization method with the boundary variations method is a problem of generating outlines of objects from grey level pictures. A procedure for an automatic computation of the proposed initial form for the boundary variations technique could thus be based on ideas and techniques from image analysis and pattern recognition. For the examples presented in this paper, the outlines for the initial proposed form were generated manually thus mimicking a design situation where the ingenuity of the designer is utilized to generate a 'good' initial form from the topology optimization results. The term 'good' in this context covers considerations such as ease of production, aesthetics etc. that may not have a quantified form. A reduction of the number of holes proposed by the topology optimization by ignoring relatively small holes exemplifies design decisions that could be taken before proceeding with the boundary variations technique.

4. Boundary optimization

Once the optimal topology and initial boundary shape is defined, the objective is to refine this initial shape, such that the von–Mises equivalent stress in the body is minimized, subject to a resource and compliance constraint:

$$
\min_{\Omega \leq D} \quad \max_{x \in \Omega} \quad \overline{\sigma}_{eq}
$$

$$
\text{so:} \quad \text{Equilibrium}
$$

$$
\int d\Omega \leq V
$$

$$
\text{Compliance} \leq \overline{\phi}
$$

(6)

Here D denotes the set of admissable boundary shapes, defined through local geometric constraints.

This problem is solved with a gradient technique, with shape sensitivities obtained via the speed method ([5]). For the sensitivity analysis, very precise estimates of stress is required and for this reason, the equilibrium is defined via the stationarity condition for the Hu–Washizu variational principle ([8]). Also, a boundary fitted elliptic mesh generator is used to generate the FEM–mesh used for the numerical solving procedure for the mixed analysis problem. This mesh generator is employed at each iteration step of the boundary optimization, thus maintaining good mesh properties throughout the shape modification process [5]. In order to cater for the non–simply connected domains predicted by the topology optimization system, the mesh generator is based on a subdivision of the domain by blocks. The remeshing is a crucial element in the boundary optimization procedure and together with the use of a mixed FEM method, allows for the boundary movements to be parametrized by movement of the FEM nodes along the design boundaries ([8]).

5. Examples

Figures 2 and 3 show examples of 2–dimensional structures optimized through the material distribution method followed by the boundary variations technique, as described above.

As can be seen, the topology optimization results in very good initial forms obtained for the boundary variations technique. Generally, only small and localized design changes occur during the boundary optimization.

Typically, the minimization of the stress level during the boundary optimization also results in some decrease in the compliance, but this is not unexpected as the drawing of the initial form from the topology data constitutes a not insignificant pertubation of the minimum compliance design.

References

[1] R.T. Haftka and R.V, Gandhi: Structural shape optimization — A survey. Comp. Meths. Appl. Mech. Engrg. 57 (1986) 91—106.

[2] M.P. Bendsøe and N. Kikuchi: Generating optimal topologies in structural design using a homogenization method. Comput. Meths. Appl. Mechs. Engrg. 71 (1988), 197—224.

[3] M.P. Bendsøe: Shape design as a material distribution problem. Struct. Optim., 1 (1989), 193—202.

[4] Suzuki, N., Kikuchi, N.: Shape and Topology Optimization for Generalized Layout Problems Using the Homogenization Method. Comp. Meths. Appl. Mechs. Engrg. (to appear).

[5] M.P. Bendsøe, and H.C. Rodrigues: Integrated topology and boundary shape optimization of 2—D solids. Comp. Meths. Appl. Mechs. Engrg. (to appear).

[6] Bremicker, M.; Chirehdast, M.; Kikuchi, N.; Papalambros, P.: Integrated Topology and Shape Optimization in Structural Design. Computers and Structures. (to appear).

[7] P. Pedersen: On optimal orientation of orthotropic materials. Struct. Optim., 1(1989), 101—106.

[8] H. Rodrigues: Shape optimal design of elastic bodies using a mixed variational formulation. Comp. Meths. Appl. Mechs. Engrg. 69 (1988) 29—44.

MACRO-SCALE

MICRO-SCALE 1

MICRO-SCALE 2
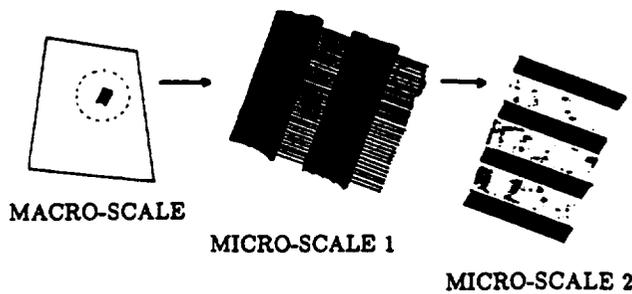
FIG.1. Construction of a layering of second rank.

a

b

c

FIG.2. Optimal design of a beam. A: Optimal topology with outline showing reference domain. B and C: Initial and final design using the boundary variations method. Two blocks are used for the elliptic mesh generator. Only the boundaries of block 1 can move. The maximum stress is reduced by 55.7% and the compliance by 7.3%. Block divisions are shown as hatched, bold lines: design boundaries as bold solid lines.
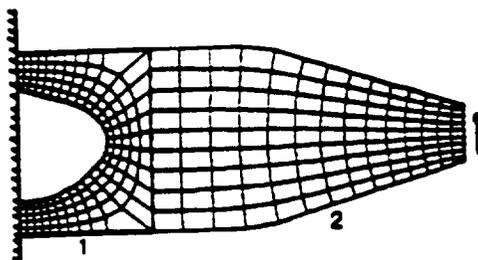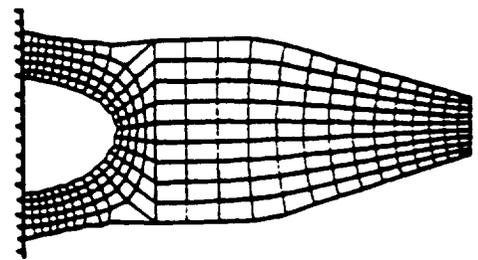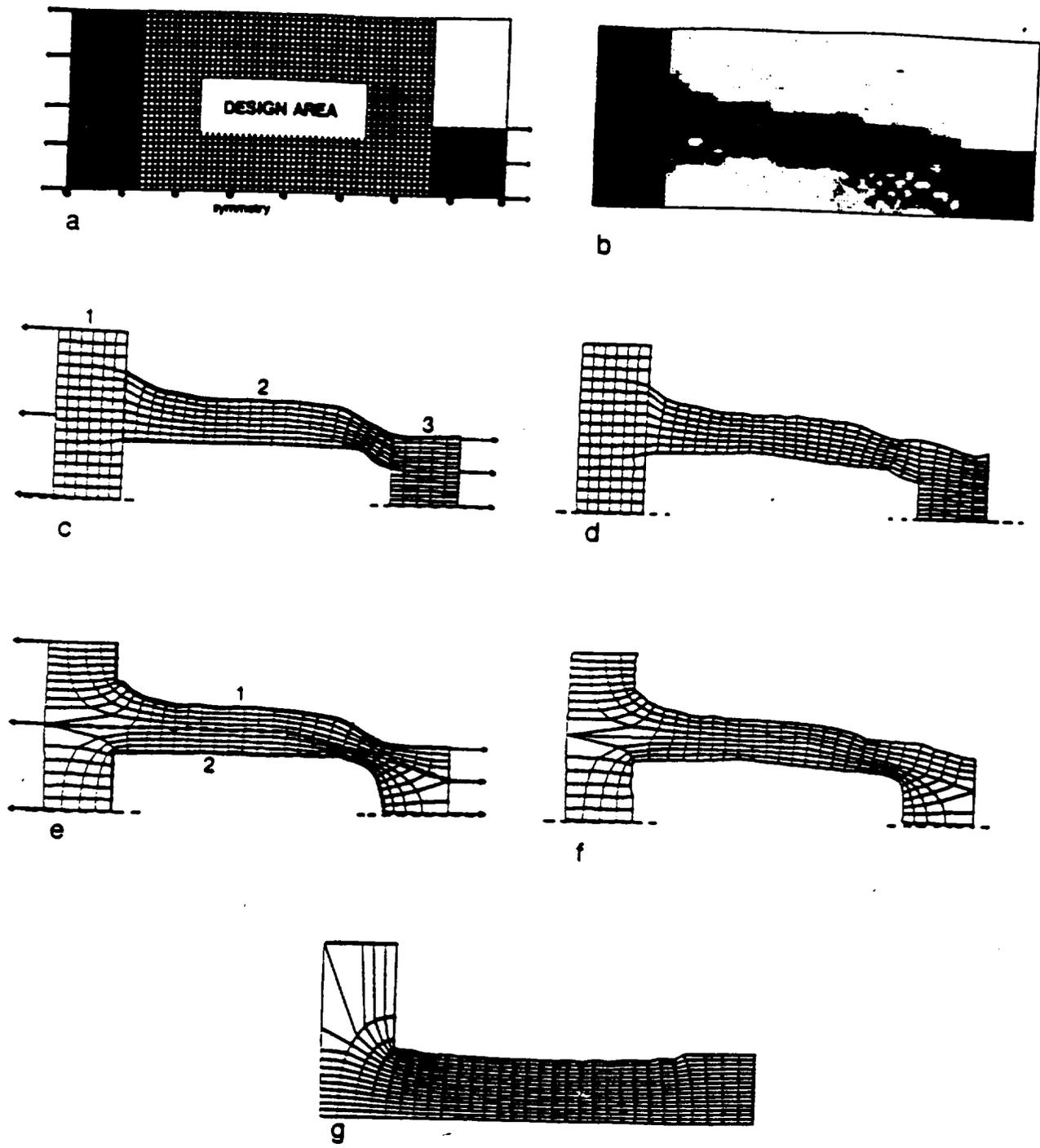
FIG.3. Optimal design of a fillet. A: The reference domain showing loads etc. and the reduced design area. B: The optimal topology. C and D: Initial design and optimal design with the boundary variations method. Mesh generated using three blocks. E and F: Initial and final optimal design using different mesh (two blocks). G: Optimal design with no hole (minimum compliance design).

CRYOGENIC OPTICAL ASSEMBLY (COA) COOLDOWN ANALYSIS FOR THE COSMIC BACKGROUND EXPLORER (COBE)

Robert J. Coladonato
Sandra M. Irish
Carol L. Mosier

NASA Goddard Space Flight Center
Greenbelt, MD 20771

N94-71467

INTRODUCTION

The Cosmic Background Explorer (COBE) Spacecraft, developed by Goddard Space Flight Center (GSFC), was successfully launched on November 18, 1989 aboard a Delta Expendable Launch Vehicle. Two of the three instruments for this mission were mounted inside a liquid helium (LHe) dewar which operates at a temperature of 2 Kelvin (K). These two instruments are the Diffuse Infrared Background Experiment (DIRBE), and the Far Infrared Absolute Spectrophotometer (FIRAS). They are mounted to a common Instrument Interface Structure (IIS) and the entire assembly is called the Cryogenic Optical Assembly (COA). Figure 1 shows the dewar and COA arrangement.
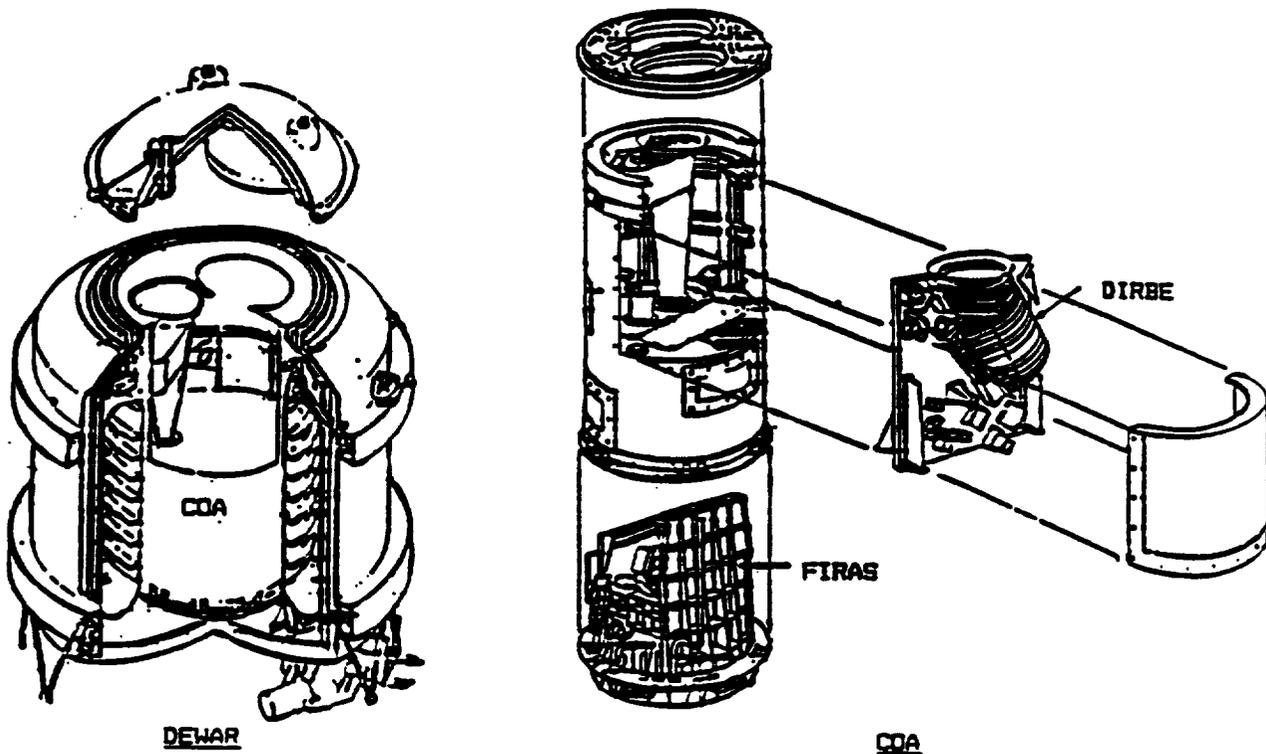
Figure 1. Dewar and COA

As part of the structural verification requirement, it was necessary to show that the entire COA exhibited adequate strength and would be capable of withstanding the launch environment. This requirement presented an unique challenge for COBE because

the COA is built and assembled at room temperature (300 K), cooled to 2 K, and then subjected to launch loads. However, strength testing of the entire COA at 2 K could not be done because of facility limitations. Therefore, it was decided to perform the strength verification of the COA by analysis.

Structural loads at 2 K needed to be combined with the launch loads to arrive at the total load. However, in order to find the loads at 2 K, possible loads from the transient cooldown event had to be considered and evaluated. A combined thermal/structural analysis was undertaken to examine the cooldown event. Transient temperature distributions and resulting structural loads due to cooldown were of concern at the three major structural interfaces: the IIS/Dewar interface, the IIS/DIRBE interface, and the IIS/FIRAS interface. Each of these interfaces is a bolted connection that relies on friction to keep the joint intact. In order for analysis to be a viable approach for strength verification it was necessary that the loads be predictable. The cooldown event had the potential for generating large temperature gradients which could result in joint slippage. Slippage at the interfaces during cooldown would introduce an unknown load condition and invalidate the analytical predictions as well as the approach for strength verification by analysis. The cooldown analysis was undertaken primarily to provide assurance that no slippage occurred; therefore, no unpredictable loads would be generated.

The combined thermal/structural cooldown analysis was a complex task. The large overall temperature range required the use of temperature dependent material properties for both the thermal and structural portions of the analysis. The temperature dependent material properties were obtained from References 1 through 7 as well as from specific in-house testing programs. The thermal analysis provided transient temperature distributions throughout the COA while the structural analysis incorporated the results from the thermal analysis and then determined interface loads.

THERMAL ANALYSIS

Both a transient and steady-state thermal analysis were conducted. The transient study determined the COA time-temperature profile during the groundhold cooldown from room (300 K) to LHe (2 K) temperatures. The dewar boundary temperatures during cooldown were provided by the Cryogenics and Propulsion Branch at the GSFC and were based on flight dewar testing at Ball Aerospace Systems Division (BASD). The steady-state temperature distribution within the COA at launch was also predicted.

THERMAL MODELS

The two types of thermal models employed in the analysis were a geometric math model (GMM) and a thermal math model (TMM). In the GMM a physical representation of the geometry was created in order to obtain radiation view factors. These view factors were included into the TMM and with conduction terms, nodal masses, dewar boundary temperatures, and material properties determined the COA nodal temperatures.

While on-orbit the entire COA is nominally below 3 K and radiation exchange is negligible. However, radiation plays a major factor in the heat transfer network for a temperature range of approximately 70 to 300 K. A geometric representation of the COA had to be generated in order to accurately model the physical heat transfer processes which occurred in this temperature range. The computer program used for this purpose was the Simplified Space Payload Thermal Analyzer (SSPTA). SSPTA (Reference 8) assumes that the radiation exchange is 100 percent diffuse. SSPTA determines configuration factors based on the model geometry. The surface properties are then introduced to obtain the radiation couplings.

The GMM contained all the major structural pieces of the IIS, DIRBE, and FIRAS and several instrument component models. Since the IIS photon cover separates DIRBE from FIRAS, the upper and lower portions of the COA could be modeled separately thereby reducing the computer run time. The IIS, FIRAS, and DIRBE geometric models

371

are shown in figure 2. Emissivity values of 0.75 and 0.10 were assumed for all black anodized and bare aluminum structure, respectively. Based on the results of a preliminary cooldown analysis (December 1983) most of the primary structure was black anodized to enhance radiation exchange and thereby reduce gradients.

DIRBE OPTICS PLATE

IIS STRUCTURE   FIRAS STRUCTURE
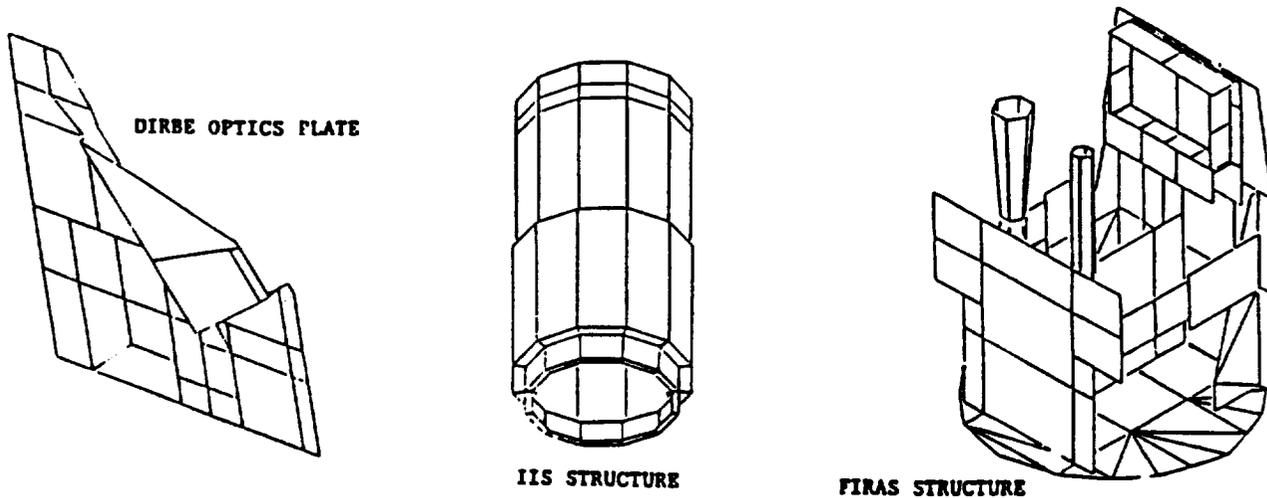
Figure 2. Thermal Geometric Models

The approximately 400 node TMM, which was used to predict temperatures, was generated in the Systems Improved Numerical Differencing Analyzer (SINDA) format. SINDA (Reference 9) is a finite differencing program which defines a thermal network, manipulates solution arithmetics, and specifies output of results. The program requires the user to specify a network of thermal modeling elements unique to the physical problem. These lumped parameters include nodal thermal masses, conductances, internal power generation values, radiation couplings, and boundary temperatures. In cryogenic modeling the conductivity and specific heat of materials are highly temperature dependent. Interface conductances are also functions of temperature. Values, determined by testing, are stored in arrays which reference them to the associated temperatures. The correct material properties are determined during each iteration by linear interpolation of entries in the arrays. Conductances of critical paths were verified during COA cryogenic testing.

The solution network routine specifies the type of finite differencing technique and convergence criteria to be used. For the steady-state analysis "STDSTL" was chosen. In this routine the finite difference equations are solved iteratively by the Gauss-Seidel method. For the transient cooldown analysis "FWDBCK" was used. This routine uses a forward-backward differencing technique with the general quartic formula. Convergence for both of these routines was based on both a temperature relaxation criteria and a desired energy balance with maximum values of 0.5 mK and 0.1 mW, respectively.

THERMAL RESULTS

Results for the cooldown analysis were generated in the form of time-temperature profiles of all SINDA nodes. Of particular interest was the gradients at the IIS/Dewar, IIS/DIRBE, and IIS/FIRAS interfaces. Figure 3 shows the average temperature gradients across these interfaces. The cooldown profiles, along with steady-state prelaunch temperatures, were used as input for the NASA Structural Analysis (NASTRAN) study.

372
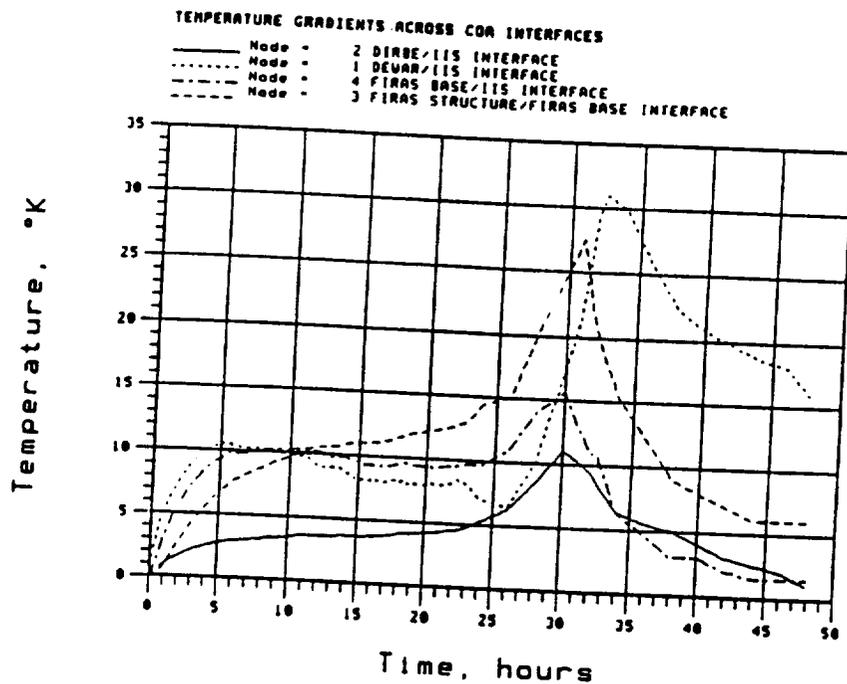
Figure 3. Average Temperature Gradients Across COA Interfaces

STRUCTURAL ANALYSIS

NASTRAN MODELS

NASTRAN is a computer program which uses the finite element method to solve a wide variety of engineering problems. In order to use this software, a finite element model of the structure in question is required. NASTRAN models of the COBE dewar and COBE instruments were obtained to perform the structural portion of the cooldown analysis. These models were acquired from the Cryogenics and Propulsion Branch and the Electromechanical Branch at the GSFC. Figure 4 is a plot of the NASTRAN models utilized in the analysis. Only the internal tank, suspension straps, and support brackets of the dewar model were needed for the analysis.

Since the models were developed from different organizations, they were formatted in dissimilar versions of NASTRAN. The models had to be converted into the Universal Analytics Incorporated (UAI) version of NASTRAN since a UAI/NASTRAN (Reference 10) model coupling technique was used in the analysis. This technique, known as substructuring, allows for the coupling of many detailed models. It permits the analyst to work with smaller, more manageable components rather than a very large combined model. The models, if combined directly to form one NASTRAN model, would have contained 40,000 degrees of freedom (DOF). By using this technique, the results were obtained in less computer time than if one large NASTRAN model of the COA structure had been analyzed.

Once each model was converted into UAI/NASTRAN, elements such as bar offsets or rigid elements were removed since they do not allow proper thermal expansion. The rigid elements were replaced by stiff, elastic bars that would expand freely under thermal loading. Checks were made to assure matrix stiffness ratios were not exceeded. Also, thermal check runs were performed on each model to verify proper thermal expansion, rigid body motion, stiffness equilibrium and adequate geometry of elements. The final check was to verify that each model contained the appropriate grids for connection to the other models.
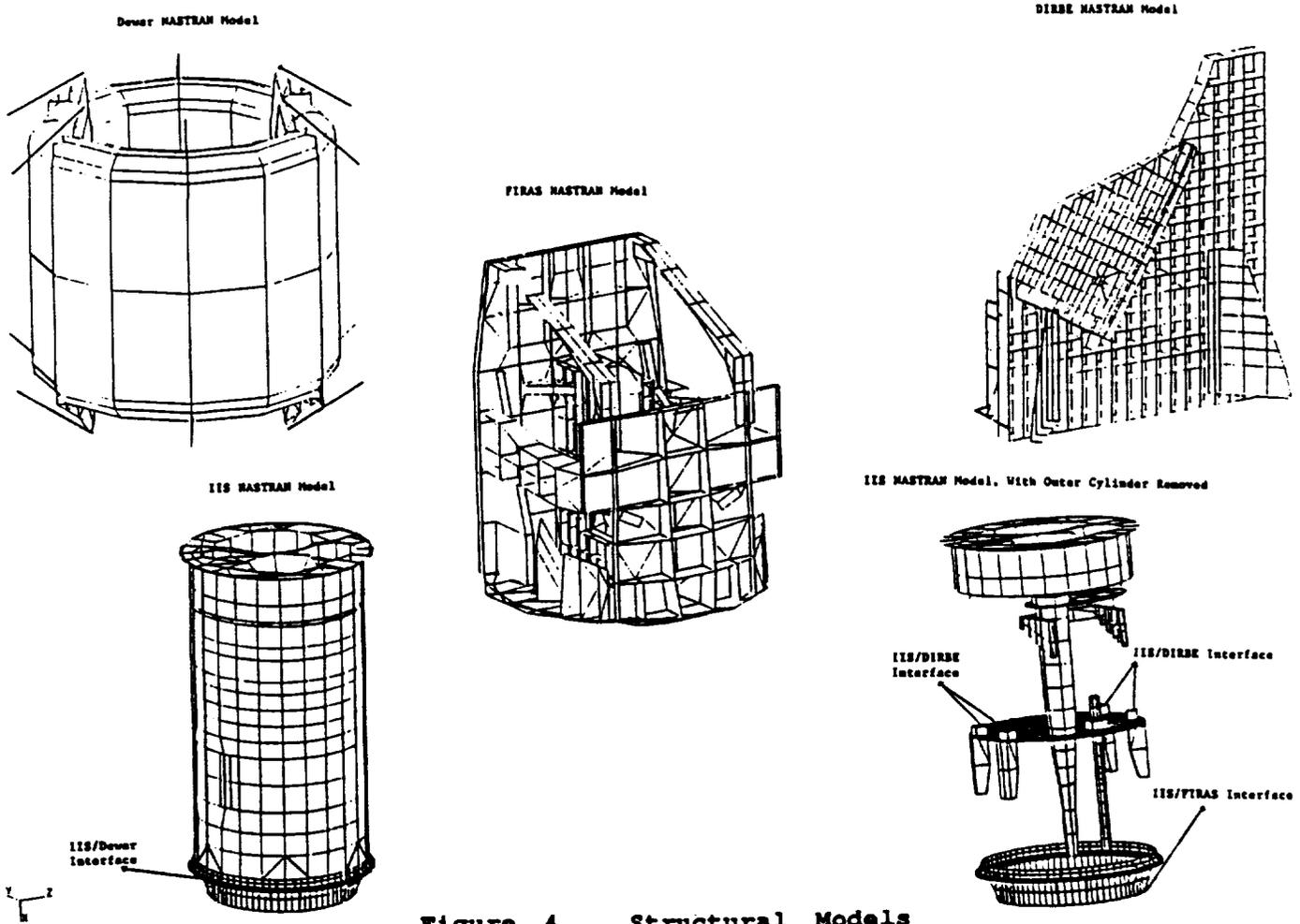
Figure 4.   Structural Models

APPLYING TEMPERATURE DATA

Time-temperature profiles from the thermal analysis were provided to the structures group. These profiles were utilized to predict the appropriate time steps to perform the structural analysis. For this analysis, the temperature distributions at the time steps of 5.5 hours, 6.083 hours, 19.25 hours, 28.5 hours, 30.5 hours, and 34.5 hours after start of cooldown were applied to the combined COA model. These time steps were chosen by reviewing the temperature gradients across the interfaces and taking into account the thermal expansion coefficients of each material as it varies with temperature. In addition to the six time steps chosen, one case was run which applied a temperature of 2 K on the entire structure. This run was used to check the interface loads that remain after the completion of cooldown.

To properly apply the predicted temperatures to the structural models, a direct mapping of the SINDA thermal nodes to the NASTRAN structural grids was performed. The geometric equivalent structural grid was assigned to each thermal node and then the node's temperature was applied to that grid. The thermal and structure groups worked together to select and assign the mapping equivalence.

A FORTRAN program was developed which simplified the procedure of reading the temperature data, searching for the corresponding grid, applying the temperature data to the grid, and formatting the information into TEMP cards which could be read directly by the NASTRAN Thermal Analyzer (NTA). The NTA (Reference 11) was used to interpolate between known temperatures and calculate the temperatures of all the other grids in the structural model. Non-linear temperature interpolation was

374

chosen since the thermal conductivities of each material were input as a function of temperature. The NTA produced a temperature distribution for each time step. Color graphics plots were developed to obtain a better understanding of the temperature distributions applied to the models. These plots were created by using the PATRAN pre- and post-processing software on a Silicon Graphics IRIS Workstation.

INTERFACE LOADS COMPUTATION

Once all the temperature distributions were known from the NTA, the models were arranged into UAI/NASTRAN substructuring runs to obtain the loads at the IIS/Dewar, IIS/DIRBE, and IIS/FIRAS interfaces. All bolted locations were represented by a three translational DOF connection, except the dewar internal tank interface to the dewar external tank which was fixed in all six DOF. Also, all material properties used in the analysis were input as a function of temperature.

Table 1 lists the maximum interface loads obtained by NASTRAN during cooldown for each interface. The maximum loads were computed as the root-sum-squared of the two lateral loads for each bolt. Table 1 also indicates the time step during cooldown that the maximum load occurred and the temperatures on either side of the interface. The maximum loads for the IIS/DIRBE and IIS/FIRAS interfaces are low whereas the maximum load for the IIS/Dewar interface is substantially higher. This is primarily due to a large temperature gradient being applied to an interface which has large stiffness and contains slightly different materials. The dewar is Aluminum 5083 and the IIS is Aluminum 6061. A plot of the maximum allowable interface loads versus temperature for the IIS/Dewar, IIS/DIRBE, and IIS/FIRAS interfaces is shown in figure 5. With this figure and the loads presented in table 1, it is clear that the loads calculated from the analysis at the IIS/Dewar, IIS/DIRBE, and IIS/FIRAS interfaces are within the allowable load. Table 1 also presents the maximum interface loads for the case where the entire structure was held at 2 K equilibrium. These loads were found to be quite low and they did not present a problem.

Table 1. Maximum Interface Loads

INTERFACE	MAXIMUM LOAD (lb)	TIME AT MAX. LOAD	TEMPERATURE (Kelvin)	2 KELVIN LOAD (lb)
IIS/Dewar	1236	6.083 Hr	IIS=241.8 Dewar=225.4	159
IIS/DIRBE	300	28.5 Hr	IIS=77.9 DIRBE=90.9	10
IIS/FIRAS	483	6.083 Hr	IIS=244.3 FIRAS=255.2	1
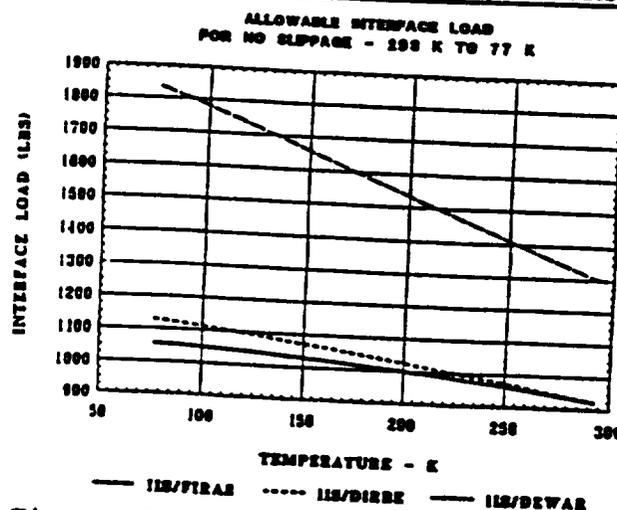
Figure 5. Allowable Interface Loads

Several cooldowns of the COA were done, starting with the first in April 1985 and culminating with the cooldown of the flight COA in the flight dewar in August 1989. The first cooldown was done using a Thermal Structural Unit (TSU) of the COA in a LHe Instrument Test Dewar (ITD). The IIS/Dewar interface was instrumented with temperature sensors for all cooldowns and additionally, for the first cooldown, the TSU/COA was instrumented with additional temperature sensors and strain gages. The intent for this first test was to follow the nominal cooldown curve that had been used in the analysis. However, the initial cooldown rate proceeded more rapidly than anticipated. After about 16 hours the cooldown was stabilized for a long period before continuing. Examination of the data at the IIS/Dewar interface showed a maximum temperature gradient that would result in the predicted interface load exceeding the allowable to prevent slippage by 10 percent. Locations that were instrumented with strain gages showed stress levels around 2000 pounds per square inch (psi) compared to the model predictions of approximately 3000 psi. Upon removal of the COA from the ITD, a thorough visual examination showed no signs of structural damage or evidence of slippage at the IIS/Dewar interface. Based on the operational experience learned from the first cooldown, enhanced procedures and methods were adopted and all subsequent cooldowns in the ITD or flight dewar were performed at rates that were less than the nominal rate used for the analysis. Figure 6 shows the cooldown curves for the April 1985 ITD cooldown, the analysis cooldown, and the August 1989 flight dewar cooldown. The plot on the left shows the first 10 hours and the plot on the right shows the total time. Data obtained from the temperature sensors during the final August 1989 cooldown showed that the temperature gradient at the IIS/Dewar interface was an order of magnitude below the allowable gradient which would cause slippage at the interface.
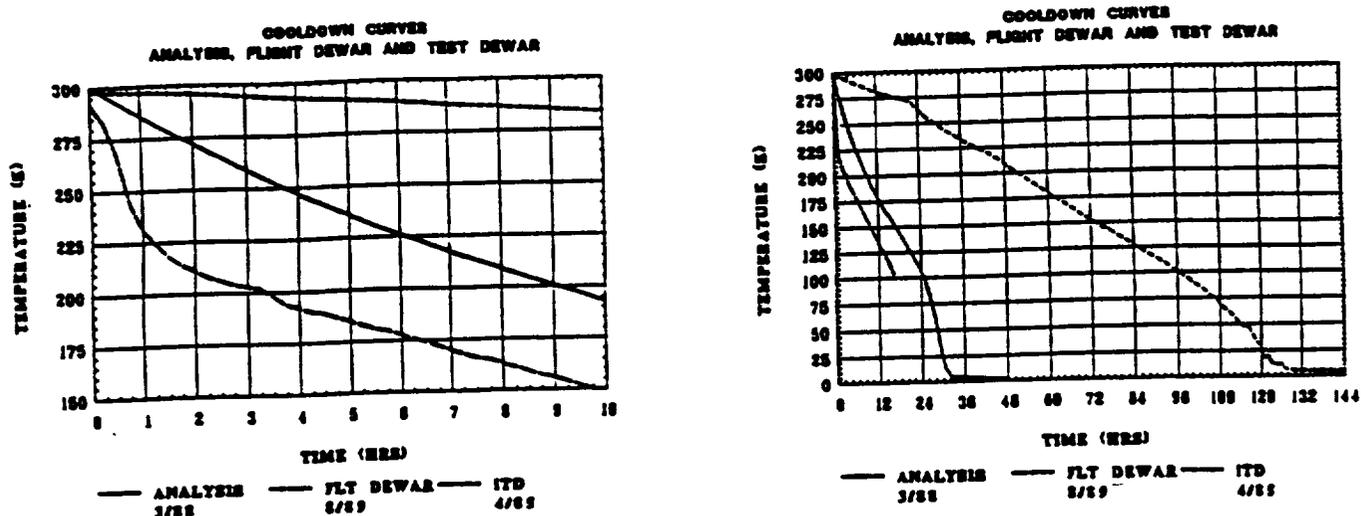
Figure 6. Cooldown Curves

CONCLUSION

The cooldown analysis predicted and identified potential problem conditions and areas related to the structural verification program for the COA. Based on the results of the cooldown analysis, steps were taken to better control the cooldown process which in turn minimized induced temperature gradients and interface loads. The very slow cooldown of the flight COA in the flight dewar provided assurance that no unpredicted loads would be present during launch. Under this condition, strength verification by analysis was considered to be an acceptable approach.

LIST OF ACRONYMS

BASD	Ball Aerospace Systems Division
COA	Cryogenic Optical Assembly
COBE	Cosmic Background Explorer
DIRBE	Diffuse Infrared Background Experiment
DOF	Degrees of Freedom
FIRAS	Far Infrared Absolute Spectrophotometer
GMM	Geometric Math Model
GSFC	Goddard Space Flight Center
IIS	Instrument Interface Structure
ITD	Instrument Test Dewar
K	Kelvin
LHe	Liquid Helium
LN2	Liquid Nitrogen
NASA	National Aeronautics and Space Administration
NASTRAN	NASA Structural Analysis
NTA	NASTRAN Thermal Analyzer
PSI	Pounds Per Square Inch
SSPTA	Simplified Space Payload Thermal Analyzer
SINDA	Systems Improved Numerical Differencing Analyzer
TMM	Thermal Math Model
TSU	Thermal Structural Unit
UAI	Universal Analytics Incorporated

REFERENCES

1. Cryogenic Materials Data Handbook, Volume II, Air Force Materials Laboratory, Technical Documentary Report, AFML-TDR-64-280, July 1970.

2. Handbook on Materials for Superconducting Machinery, Metals and Ceramic Information Center, MCIC-HB-04, November 1974.

3. Clark, Alan F. and Richard P. Reed, eds., "Materials at Low Temperature," American Society for Metals, National Bureau of Standards, 1983.

4. Westinghouse Micarta Cryogenic Materials Data Sheet, G-10CR, (Micarta 56G12).

5. Beekman, D. H. Jr., M. B. Kasen, G. R. MacDonald, R. E. Schramm, Mechanical, Electrical, and Thermal Characterization of G-10CR and G-11CR Glass-Cloth/Epoxy Laminates Between Room Temperature and 4K, National Bureau of Standards, Boulder, Colorado.

6. Handbook of Physics and Chemistry, 59th Edition, 1978-1979.

7. Overton, W. C. Jr., C.L. Tsai, H. Weinstock, "Low Temperature Thermal Conductivity of Stycast 2850FT," Cryogenics Journal, September 1978.

8. Program Manual for the Simplified Space Payload Thermal Analyzer, Prepared by Arthur D. Little, Inc., Cambridge Massachusetts 02140, October 1986, Contract NAS5-27606.

9. SINDA User's Manual, Prepared by Lockheed Engineering and Management Services Company, Inc., Houston, Texas, Revision 3, March 1983, Contract NAS 9-15800.

10. UAI/NASTRAN User's Manual, Volume I and II, Version 9.6, January 1988.

11. Jackson, Clifton E. Jr., and Hwa-Ping Lee, "NASTRAN Thermal Analyzer-Theory and Application Including a Guide to Modeling Engineering Problems, Volumes I and II," NASA TM X-3503 and X-3504, April 1977.

DESIGN OPTIMIZATION AND PROBABILISTIC ANALYSIS OF A HYDRODYNAMIC JOURNAL BEARING

ALEXANDER G. LINIECKI
MECHANICAL ENGINEERING DEPARTMENT
SAN JOSE STATE UNIVERSITY

Abstract

A nonlinear constrained optimization of a hydrodynamic bearing was performed yielding three main variables: radial clearance, bearing length to diameter ratio and lubricating oil viscosity. As an objective function a combined model of temperature rise and oil supply has been adopted. The optimized model of the bearing has been simulated for population of 1000 cases using Monte Carlo statistical method. It appeared that the so called "optimal solution" generated more than 50% of failed bearings, because their minimum oil film thickness violated stipulated minimum constraint value. As a remedy change of oil viscosity is suggested after several sensitivities of variables have been investigated.

There were many papers devoted to journal bearing optimization. Authors like: Seireg, Ezzat, Bartel, Marks and others [1, 3, 6] developed various numerical techniques in order to minimize some objective function involving heat generation, bearing size, temperature rise, etc.

Bearing design problem is the representative of a large class of engineering design problems and includes many behavior and design variables related through highly nonlinear and complex relationships. The performance of modern high speed rotating machinery depends to a large extent on properly designed journal bearings. Each one of them presents a complex thermo-mechanical system in which there exist many competing objectives. Using computer aided automated search methods we encounter many engineering constraints which should not be violated. One of the very well known techniques is the numerical optimization based on equations developed by curve fitting the numerical solutions of various parameters from Raimondi and Boyd's numerical solution of the famous Reynold's equation of hydrodynamic bearing. This numerical solution based on non-dimensional variables established a universal model serving the solution of thousands of different configurations of journal bearings. The automated search for optimal design used mainly deterministic models and applied nonlinear programming techniques with various complex nonlinear constraints. However, the stochastic behavior of the optimized bearing model has been unknown.

This paper represents an attempt to confront the numerically optimized journal bearing model with its stochastic behavior. It shows the futility of the so called "optimal design" if the probabilistic variability of the main parameters is ignored. As an example, a journal bearing has been selected with a fixed constant load W = 3000 lbs., nominal diameter D = 4 in. and rotational speed at N = 80 rev/sec. Three design independent variables were defined:

1. X_1 = L/D - length to diameter ratio
2. $X_2 = C_r$ - radial clearance
3. $X_3 = \mu$ - absolute viscosity of lubricating oil

There are several objectives which can be pursued in bearing design:

1. Minimum oil temperature rise in the bearing,
2. Minimum oil flow rate required for adequate lubrication,
3. Minimum power loss in the bearing,
4. Maximum of the minimum oil film thickness.

In this case the criterion for optimum design has been selected as a combination of two competing objectives 1 and 2.

The objective function model $F = \Delta T + 8Q$ includes minimum oil temperature rise ΔT in °F due to viscous friction and minimum quantity of oil Q in³/sec supplied to bearing due to hydrodynamic leakage with a weighting factor of 8.

The objective function presents a tradeoff between conflicting tendencies of oil flow and temperature growth. The following engineering constraints have been selected:

1. $h_o \geq .00035$ in where: h_o is the minimum oil film thickness,
2. $T \leq 200°$ F where: T is the highest temperature of the lubricating oil in the bearing,
3. $C_r \geq .002$ in. where:

$$C_T = \frac{D_B - D_S}{2}$$

D_B - bearing diameter
D_S - shaft diameter

4. $.25 \leq L/D \leq .50$ where:

    L - bearing length
    D - bearing nominal diameter

5. $P_{max} \leq 4000$ psi where: P_{max} is the highest oil pressure developed in the lubricating oil film,

6. $\mu \geq .5E\text{-}6$ reyns where: μ is the absolute viscosity of the lubricating oil film

7. $h_o \leq .15\ C_r$ for adequate loading of the bearing

8. $.15 \geq S \geq .01$ where: S is the Sommerfeld number, a nondimensional reference variable in the form:

$$S = \left(\frac{D}{2C_r}\right)^2 \left(\frac{\mu N}{P}\right)$$

where the new parameter P is the load per unit of projected bearing area:

$$P = \frac{W}{DL} \quad \text{(psi)}$$

There are 6 other constraints of minor importance.

Seireg and Ezzat [1] presented a series of curve fits for the full 360° bearing as :

$$h_o = 1.585 C_r\ (L/D)^{.913}\ (S)^{.655(L/D)^{.0922}} \quad \text{(inch)}$$

for minimum oil film thickness,

$$\Delta T = .5/(L/D)^{.374}\ (P)(S)^{.695/(L/D)^{.139}} \quad (°F)$$

for temperature rise in the oil film,

$$P_{max} = P/.76\ (L/D)^{.62}\ (S)^{.24} \quad \text{(psi)}$$

for maximum pressure in the oil film,

$$Q = DNC_rL/.256\ (L/D)^{.048}\ (S)^{.1(L/D)^{.47}} \quad \text{in}^3/\text{sec}$$

for the quantity of oil fed to bearing.

These equations are valid for the range of Sommerfeld Nr in the range:

$$0 \leq S \leq .15$$

The objective function and all constraints are extremely nonlinear. The optimization process was performed using numerical code based on the Hooke-Jeeves pattern search method. This method which includes step size acceleration-deceleration and flexible penalty functions has been developed by C. W. Radcliffe at U. C. Berkeley.

The initial guesses of independent variables were:

$$X_1 = L/D = .300$$
$$X_2 = C_r = .002 \text{ in.}$$
$$X_3 = \mu = 1.E\text{-}6 \text{ reyns}$$

with objective function evaluated as:

$$F = \Delta T + 8Q = 119.13$$

Here temperature rise was:

$$\Delta T = 90.55^\circ F$$

and oil supply:

$$Q = 3.64 \text{ in}^3/\text{sec}$$

After 629 function evaluations the converged solution reduced the initial value of F and established independent variables as:

$$F = 92.66$$
$$L/D = .3694$$
$$C_r = .002836 \text{ in.}$$
$$\mu = .7997E\text{-}6 \text{ reyns}$$

The oil temperature rise has been reduced to:

$$\Delta T = 40.38^\circ F$$

but the oil flow has been increased to:

$$Q = 6.535 \text{ in}^3/\text{sec}$$

None of the 14 constraints has been violated.

The most sensitive parameter in bearing design is its minimum oil film thickness h_o. If it is too small a high risk of bearing failure exists. For optimal design the value of h_o came out as:

$$h_o = .00035 \text{ in. with eccentricity ratio:}$$
$$\varepsilon = .8766$$

The optimization process using deterministic single valued approach has been formally concluded. Now the second important procedure has been applied, namely: the optimized bearing was tested for its stochastic behavior assuming a large series of bearings in the production plan.

For the population of 1000 bearings a computer aided Monte Carlo simulation has been performed in order to obtain the variability distribution within the set of accepted tolerances.

A Gaussian (Normal) distribution of the variables has been assumed for the optimal design case. The following tolerances have been established:

381

$$L/D = \mu_{L/D} \pm 5\% = \mu_{L/D} \pm 3\sigma_{L/D}$$

$$C_r = \mu_{C_r} \pm 10\% = \mu_{C_r} \pm 3\sigma_{C_r}$$

$$\mu = \mu_{mean} \pm 10\% = \mu_{mean} \pm 3\sigma_{\mu}$$

Each tolerance field encompassed 6 standard deviations which translated into numbers gave the following normally distributed set of independent variables at optimal solution:

$$X_1 = L/D = N(\mu_{L/D}; \sigma_{L/D}) = N(.3694; .006156)$$

$$X_2 = C_r = N(\mu_{C_r}; G_{C_r}) = N(.00283; 9.453E\text{-}5)$$

$$X_3 = \mu = N(\mu_{mean}; G_{\mu}) = N(.799655E\text{-}6; .26655E\text{-}7)$$

At the same time minimum oil film thickness h_o became a stochastic function of above mentioned variables: X_1, X_2, X_3.

$$h_o = 1.585 X_2 X_1^{.913} (S)^{.655 X_1^{.0922}} \qquad \text{where:}$$

$$S = \left(\frac{D}{2X_2}\right)^2 \left(\frac{X_3 N}{P}\right)$$

For each variable two independent random numbers $R_i^{(1)}$, $R_i^{(2)}$ were generated 1000 times in the ranges:

$$0 < R_i^{(1)} < 1$$
$$0 < R_i^{(2)} < 1$$

and then used in each variable X_i normally distributed. From Gaussian probability density function for variable X_i:

$$f(X_i) = \frac{1}{\sigma_i \sqrt{2\pi}} e^{-\frac{(X_i - \mu_i)^2}{2\sigma_i^2}}$$

the inverse transformation for X_i has been used as:

$$X_i = \mu_i + \sqrt{-2 \ln R_i^{(1)}} \cos (2\pi R_i^{(2)}) \sigma_i \qquad \text{where:}$$

μ_i - mean value for i-th variable.

A histogram of h_o was generated from the population of 1000 cases of produced bearings (figure 1).
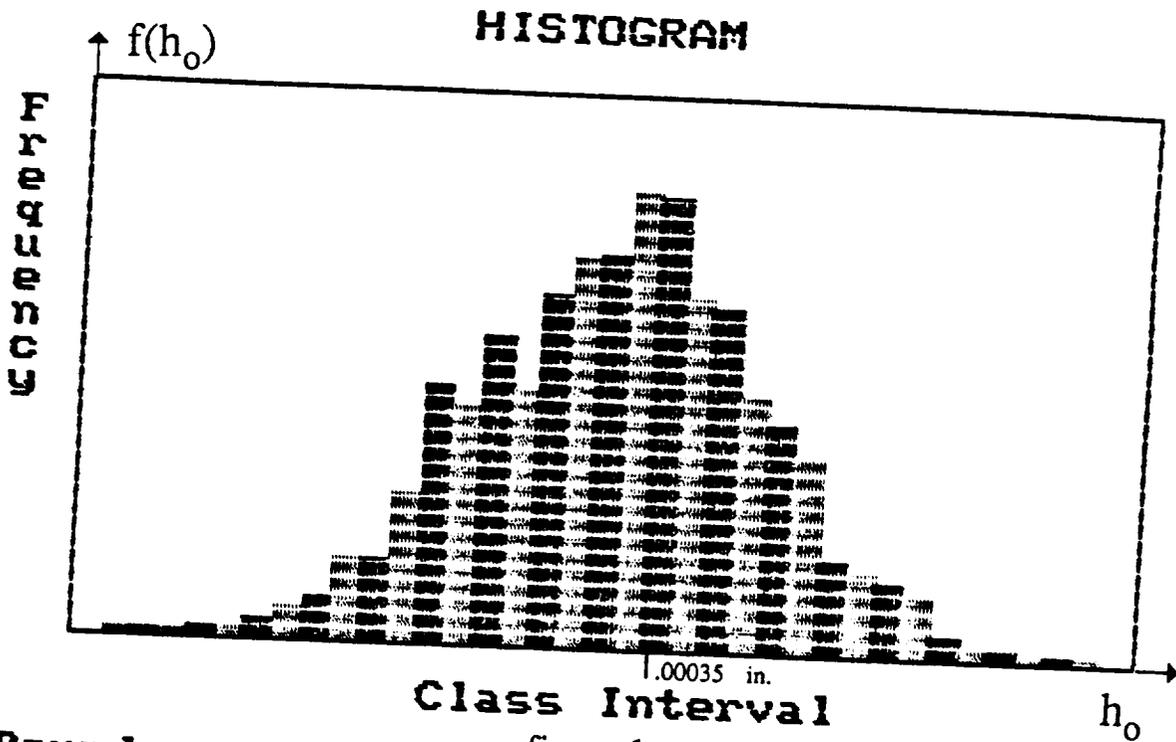
HISTOGRAM

figure 1

Bounds on X
Lower = 3.093914E-04
Upper = 3.842646E-04

Press ◄____

From this histogram the range for h_o is:

$$3.0939E\text{-}4 \le h_o \le 3.8426E\text{-}4 \quad inch$$

from which it is obvious that more than 50% of the produced bearings will have to be rejected because the most important constraint:

$$h_o \ge .00035 \quad \text{has been violated that many times.}$$

That would produce an economic disaster for the bearing manufacturer. The improvement of the h_o distribution has been achieved by modification of one of three independent variables: X_1, X_2, X_3 for which the sensitivity of h_o was the highest.

At the optimum point for objective function:

$$F_{min} = \Delta T + 8Q$$

sensitivities were numerically determined as:

$$\frac{\partial h_o}{\partial (L/D)} = 1.4710E\text{-}04$$

$$\frac{\partial h_o}{\partial (C_r)} = -2.2914E\text{-}02$$

$$\frac{\partial h_o}{\partial (\mu)} = 327.8$$

It became obvious that the selected oil viscosity affects mostly the of h_o population of 1000 bearings. Required correction of the oil viscosity can be found from the simple relationship obtained from the first term of Taylor expansion:

$$\Delta h_o = \frac{\partial h_o}{\partial \mu} \Delta \mu \quad \text{so that}$$

$$\Delta \mu = \Delta h_o \Big/ \frac{\partial h_o}{\partial \mu}$$

Here the value of correction Δh_o is found from the difference:

$$\Delta h_o = 3.5000E\text{-}4 - h_{o(low)}$$

selecting new low value for h_o as:

$$h_{o(low)} = 3.2000E\text{-}4 \quad \text{the oil film thickness correction is:}$$

$$\Delta h_o = 3.5000E\text{-}4 - 3.2000E\text{-}4 = .3000E\text{-}4 \quad \text{inch}$$

and correction in viscosity of oil:

$$\Delta \mu = \frac{.3000E\text{-}4}{327.8} = 9.1463E\text{-}8 \text{ reyns}$$

The new mean oil viscosity became:

$$\mu_{new} = \mu_{old} + \Delta \mu = 7.9970E\text{-}7 + .9146E\text{-}7 = 8.9116E\text{-}7 \text{ reyns}$$

with the standard deviation:

$$\sigma_\mu = .29705E\text{-}7 \text{ reyns}$$

Once again performed Monte Carlo simulation revealed shifting of the range for distribution of h_o significantly to the right (figure 2) so that:

$$3.36690E\text{-}4 \leq h_o \leq 4.09534E\text{-}4 \quad \text{inches}$$

with mean value:

$$h_o = 3.73740E\text{-}4 \text{ inches}$$

The new modified temperature rise is:

$$\Delta T = 39.83 \text{ °F}$$

and required oil supply

$$Q = 6.494 \text{ in}^3/\text{sec.}$$

HISTOGRAM

figure 2

Bounds on X
Lower = 3.36694E-04
Upper = 4.095339E-04

Press
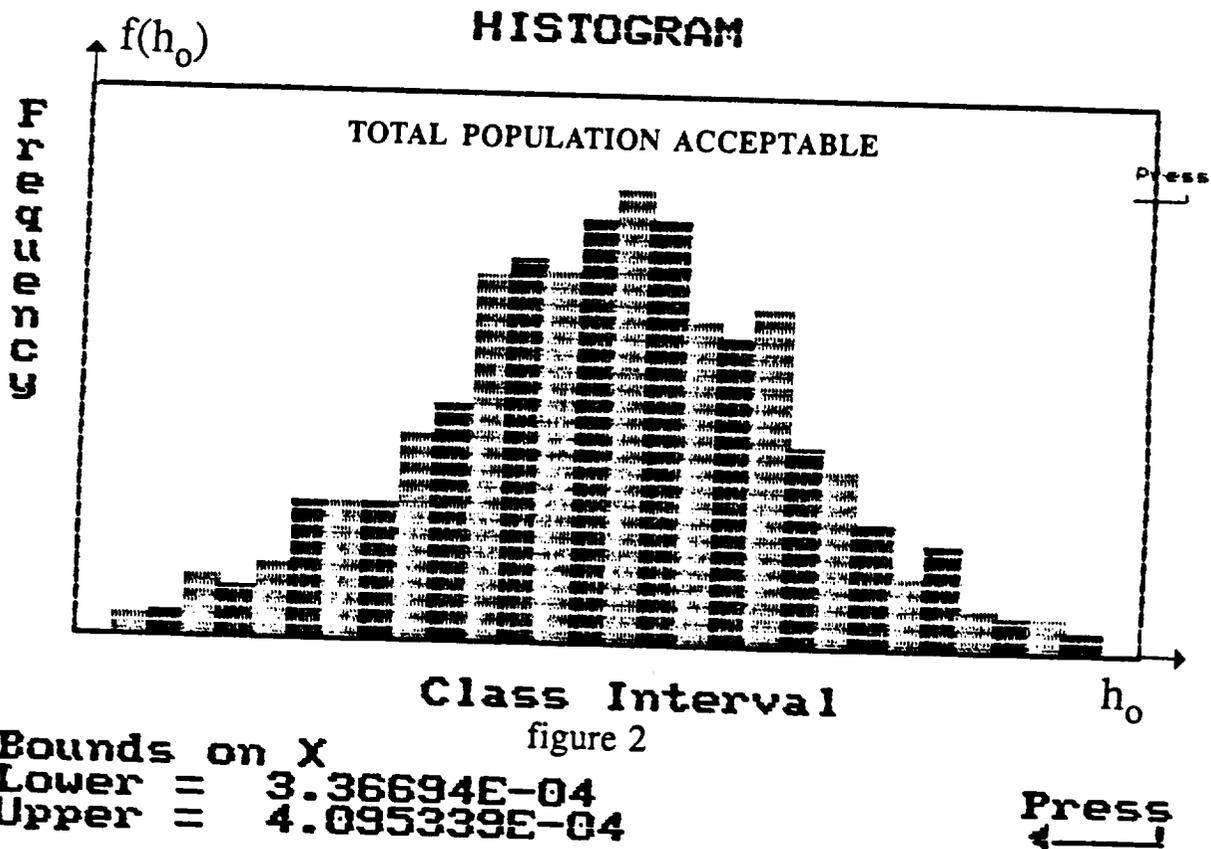
Results of numerical optimization and stochastic simulation are displayed on Table #1:

TABLE #1

Par	Initial mean values	Optimized mean values	Stochastic Improved mean values
L/D	.3000	.3694	.3694
C_r	.00200 in	.00284 in	.00284 in
μ	1.E-06 reyns	.7997E-06 reyns	.8911E-06 reyns
ΔT	90.55°F	40.38°F	39.83°F
Q	3.57 in^3/sec.	6.53 in^3/sec.	6.50 in^3/sec
h_o	3.20E-04 in	3.50E-04 in	3.74E-04 in
S	1.280E-01	7.058E-02	6.268E-02
ϵ	.84	.88	.87

Conclusions

1. The optimization of a system based on a single case of deterministic mean values of its parameters is meaningless for a population of these systems.

2. Broad Monte Carlo stochastic simulation reveals the necessity of changing of some result parameters of optimal design for improved performance of the population.

3. Using modern computing power in two areas: optimal design and stochastic simulation, delineates a powerful new research area in design. The whole process can be fully automated.

References

1. Seireg. A. Ezzat M. "Optimum Design of Hydrodynamic Journal Bearings" Trans. ASME, *Journal of Lubrication Technology*, July 1969, pp 516-523.

2. Fox, R.L. "Optimization Methods for Engineering Design" Addison-Wesley Publishing Co. Reading, Massachusetts, 1971.

3. Bartel, D.L., Marks, R.W., "The Optimum Design of Mechanical Systems With Competing Design Objectives" Trans. ASME, *Journal of Engineering for Industry*, February 1974, pp 171-178

4. Shigley, J.E., Mischke, C.R. "Mechanical Engineering Design" 5th Ed., McGraw-Hill, New York, 1989.

5. Raimondi, A.A. and Boyd, J. "A Solution for the Finite Journal Bearing and its Application to Analysis and Design" Parts I, II, and III ASME Lubrication Conference, October 1957.

6. Metwalli, S.M., Shawki, G.S.A., Mokhtar, M.O.A., Seif, M.A.A. "Multiple Design Objectives in Hydrodynamic Journal Bearing Optimization" Trans. ASME, *Journal of Mechanisms, Transmissions and Automation in Design*, 1983.

Approximate Minimum-Time Trajectories
for Two-link Flexible Manipulators

G.R. Eisler, D.J. Segalman, R.D. Robinett
Engineering Analysis Department
Sandia National Laboratories[1], Albuquerque, New Mexico

Abstract

The method of recursive quadratic programming has been used to generate approximate minimum-time tip trajectories for two-link semi-rigid and flexible manipulator movements in the horizontal plane. The manipulator is modeled with an efficient finite-element scheme for an n-link, m-joint system with bending only in the horizontal-plane. Constraints on the trajectory include boundary conditions on position and energy for a rest-to-rest maneuver, straight-line tracking between boundary positions, and motor torque limits. Trajectory comparisons utilize a change in the link stiffness to compare a semi-rigid configuration to a flexible one. The level of bending flexibility necessary to excite significant modal behavior is demonstrated. Applied torques for minimum-time maneuvers are shown to be very similiar between configurations and retain much of the qualitative character of rigid-body slewing motion.

Introduction

A variety of approaches has been advanced for rigid manipulator control, taking advantage of the fact that all or some of the controls take the form of switching functions between actuator bounds [1][2][3][4]. Pure switching functions do not lend themselves to maintaining tip accuracy for non-rigid structures. One would hope that the applied controls do take advantage of the bounds to maximize performance, but a clear analytical directive for this does not exist at the present time.

In filling this void, parameter optimization techniques can provide approximate optimal performance solutions for systems driven by complex, highly nonlinear dynamic models with arbitrary equality or inequality constraints. Of these solution techniques, the Recursive Quadratic Programming algorithm [5], embodied in the code VF02AD, has proven to be a robust tool for a variety of aerospace applications [6][7][8], and will be used in this study. The primary drawback to this or other numerical optimization methods is the dependancy on accurate gradient approximations of the performance index and constraints with respect to the parameters.

The ensuing discussion initially describes the structural dynamics model of the manipulator, followed by the optimal control problem and parameterization of the controls. It concludes with the results of a computational experiment.

The manipulator structure modeled in this study has been fabricated as a two-link, cantilever arrangement constrained to slew in the horizontal plane. Tall, thin links are used to minimize vertical plane droop. The hub or joint-1 actuator slews both links, an interlink motor, and tip payload. The interlink or joint-2 actuator located at the end of link-1 slews the second link and the tip payload. The joint-1/joint-2 actuator torque ratio used in this study is 4/1. The complete manipulator is about 0.5 meters (m) tall and 1.2m long (Fig.1).

The Structural Model

There is extensive literature discussing the difficulties of simulating the vibrations of rotating structures[9] [10] [11]. The problem arises from kinematics that are of second order importance in nonrotating problems, but become of first order importance in the presence of rotational accelerations. Additonally, there are constraints inherent to the flexible link problem which must be satisfied: motions occur entirely in a horizontal plane; one end of the chain of links is attached to a stationary hub; and each flexible link is inextensible.

The full kinematics are retained by expressing the configuration as functions of convected coordinates (i.e., coordinates attached to the arm) and measuring distance from the hub. This is a traditional approach in nonlinear elasticitity [12]. Further, the kinematic variables are selected so that all geometric constraints (fixed hub, planar motion, and non-extension) are automatically satisfied.

Since motions are assumed to occur entirely in a plane, it is also assumed that the elastic lines of the links as well as the mass centers of the cross sections all lie in the same plane. Each cross section is identified by its arc-length

distance from the hub, so that the orientation of the center of the cross section s at time t is

$$\vec{\beta}(s,t) = \cos(\theta(s,t))\vec{i} + \sin(\theta(s,t))\vec{j}$$

The vector $\vec{\beta}(s,t)$ is the unit tangent along the arm at s, and the basis vectors $\vec{i}$ and $\vec{j}$ are fixed in space.

The location of the center of cross section s (relative to a space fixed frame) at time t is obtained by integration of the above unit tangent vector:

$$\vec{x}(s,t) = \int_0^s \vec{\beta}(\hat{s},t)d\hat{s}$$

Similarly, the velocity at the cross section s at time t is obtained by integration of the time derivative of $\vec{\beta}(s,t)$:

$$\dot{\vec{x}}(s,t) = \int_0^s \dot{\theta}(\hat{s},t)\vec{\gamma}(\hat{s},t)d\hat{s} \quad \text{where} \quad \vec{\gamma}(s,t) = -\sin(\theta(s,t))\vec{i} + \cos(\theta(s,t))\vec{j}$$

The above description of configuration - entirely in terms of $\theta(s,t)$ - causes all of the geometrical constraints to be satisfied automatically. Additionally, the above description expresses the configuration in terms of one unknown field (θ), instead of the more conventional two or three fields (x, y, & θ).

The governing dynamics equations are derived using these kinematics and Hamilton's principle. A finite element discretization is used to cast the resulting integro-differential equations for $\theta(s,t)$ and its first and second derivatives into a system of fully-coupled, nonlinear algebraic equations. Since all spatial integrals are with respect to the convected coordinate, s, those integrals are configuration-independent and need be done only once. A new nonlinear system must be solved at each time step, but the quadrature process can be done in advance of the dynamics simulation.

The governing equations of motion of the are obtained from the variation of Hamilton's integrand:

$$\delta KE(t) - \delta SE(t) + \delta WE(t) = 0 \tag{1}$$

for all $t_1 \leq t \leq t_2$, where KE is kinetic energy, SE is strain energy, WE is external work.

The kinetic energy is that of the flexible links plus that of all concentrated masses and concentrated moments of inertia:

$$KE(t) = \frac{1}{2}\int_0^L \rho(s)\,\dot{\vec{x}}(s,t)\cdot\dot{\vec{x}}(s,t)ds + \frac{1}{2}\sum_{k=1}^{masses} M_k\,\dot{\vec{x}}(s_k,t)\cdot\dot{\vec{x}}(s_k,t) + \frac{1}{2}\sum_{l=1}^{inertias} I_l\,\dot{\theta}(s_l,t)\cdot\dot{\theta}(s_l,t),$$

where $\rho(s)$ is the mass per unit length measured along the length of the arm; M_k is the magnitude of the kth point mass; s_k is the convected coordinate of the kth point mass; I_l is the lth point moment of inertia; and s_l is the convected coordinate of the lth point moment of inertia.

The strain energy is that of the flexible links:

$$SE(t) = \frac{1}{2}\int_0^L \kappa(s,t)\frac{\partial\vec{\beta}(s,t)}{\partial s}\cdot\frac{\partial\vec{\beta}(s,t)}{\partial s}ds$$

where $\kappa(s,t)$ is the curvature at cross section s at time t.

The virtual work due to externally imposed torques is:

$$\delta WE(t) = \int_0^L \vec{\tau}(s,t)\cdot(\vec{\beta}(s,t)\times\delta\vec{\beta}(s,t))ds \tag{2}$$

where $\vec{\tau}(s,t)$ is the imposed torque.

Discretization along the rod of the above energy terms is obtained by discretizing the tangent vector $\vec{\beta}$ as:

$$\vec{\beta}(s,t) = \sum_{n=1}^{nodes} \vec{\beta}_n(t)p_n(s)$$

where the shape functions, p_n, are nonzero over intervals that are small relative to the anticipated radii of curvature. The shape functions used were the traditional tent-shaped basis functions, and assure compliance with the condition of nonextension. Since these functions are bases for β the resulting elements take on piece-wise circular shape, with continuous slope between the elements. Joints are defined by co-locating two nodes so that the tangent vector $\vec{\beta}$ may be discontinuous there.

388

The resulting energies are:

$$KE(t) = \frac{1}{2} \sum_{m=1}^{nodes} \sum_{n=1}^{nodes} \dot{\theta}_m(t) \dot{\theta}_n(t) \vec{\gamma}_m(t) \cdot \vec{\gamma}_n(t) M_{m,n} \ , \qquad SE(t) = \frac{1}{2} \sum_{m=1}^{nodes} \sum_{n=1}^{nodes} \vec{\beta}_m(t) \cdot \vec{\beta}_n(t) K_{m,n} \qquad (3)$$

and

$$\delta WE(t) = \sum_{m=1}^{nodes} \vec{\tau}_m(t) \cdot \vec{k} \delta \theta_m \qquad (\vec{k} = \vec{i} \times \vec{j}) \ , \qquad (4)$$

where

$$M_{m,n} = \int_0^L \rho(s) q_m(s) q_n(s) ds + \sum_{k=1}^{masses} M_k q_m(s_k) q_n(s_k) + \sum_{l=1}^{inertias} I_l p_m(s_l) p_n(s_l) \ ,$$

$$q_m(s) = \int_0^s p_m(\hat{s}) d\hat{s} \ , \qquad K_{m,n} = \int_0^L \kappa(s) p_m'(s) p_n'(s) ds \ .$$

nodes is the number of nodes, δ_K is the Kronecker delta function, $\hat{s}$ is a dummy variable, p_m' and p_n' are derivatives with respect to s of p_m and p_n, and $\vec{\tau}_m$ is the net torque applied at node m. Then the time-independent matrices, $M_{m,n}$ and $K_{m,n}$ are the topological mass and stiffness matrices, respectively.

After appropriate integration by parts, the integrand of equation 1 becomes:

$$\sum_{n=1}^{nodes} \delta\theta_m \left(-\vec{\gamma}_m(t) \cdot \ddot{\vec{\beta}}_n(t) M_{m,n} - \vec{\gamma}_m(t) \cdot \vec{\beta}_n(t) K_{m,n} + \vec{k} \cdot \vec{\tau}_n(t) \delta_k(m,n) \right) = 0 \qquad (5)$$

for all nodes m. In the above equation, τ_n is the external torque applied at node n. After $\ddot{\vec{\beta}}_n(t)$ is expanded and Equation 5 is invoked for all $\delta\theta_m$, a complete set of *nodes* second order equations in the *nodes* unknowns, $\ddot{\theta}_n$, results in:

$$\vec{k} \cdot \vec{\tau}_m(t) = \sum_{n=1}^{nodes} \left[\vec{\gamma}_m(t) \cdot \vec{\gamma}_n(t) \ddot{\theta}_n(t) M_{m,n} - \vec{\gamma}_m(t) \cdot \vec{\beta}_n(t) (\dot{\theta}_n(t))^2 M_{m,n} + \vec{\gamma}_m(t) \cdot \vec{\beta}_n(t) K_{m,n} \right] \qquad (6)$$

The above problem formulation lends itself to rapid numerical calculation. It involves only one unknown field, automatically satisfies all constraints, and requires only one evaluation of element mass and stiffness matrices Ref. [13].

Optimal Trajectory Shaping

The principal goal in this study is to combine the structural response with optimization techniques to generate actuator torque histories for accomplishing a task with minimal degradation in performance. A secondary objective is to minimize the work of a feedback controller, which will be needed to compensate for modeling errors.

A minimum-time tip trajectory was chosen for investigation. Constraints on such a trajectory include: completing a rest-to-rest maneuver, tracking a specified path $(x(t), y(t))_{tip}$, slewing between specified endpoints $[(x(t_o), y(t_o)), (x(t_f), y(t_f))]_{tip}$, and not exceeding actuator torque limits $\tau_{1,2_{max}}$.

The configuration initially starts at rest. Driving a flexible structure to rest at the final time, t_f, necessitates end constraints on both kinetic and potential or strain energies $(KE(t_f), SE(t_f))$. The chosen path is a straight line and actuator torque limits are constants. Torque limits can be integrated naturally into the controls as

$$\tau_{1,2}(t) = |\tau_{1,2_{max}}| \sin \alpha_{1,2}(t)$$

where $\alpha(t)$ is a free variable. This form assumes that the two-sided limits on $\tau_{1,2}(t)$ are of the same magnitude. Final accelerations are also to be zeroed. The problem can be restated as

minimize: $J = t_f$

subject to: - finite element model
              - input actuator torques, $\tau_{1,2}(t)$
              - known initial conditions
constrained by:

$$
\begin{bmatrix} C_j(t_f) = 0 \end{bmatrix}_{j=1,7} = \begin{bmatrix} x_{tip}(t_f) - x_{specified}(t_f) \\ y_{tip}(t_f) - y_{specified}(t_f) \\ \int_0^{t_f} \left[y_{tip}(x_{tip}(t)) - y_{line}(x_{tip}(t)) \right]^2 dt \\ KE(t_f) \\ SE(t_f) \\ \ddot{\theta}_{joint\,1}(t_f) \\ \ddot{\theta}_{joint\,2}(t_f) \end{bmatrix}
$$

Note that the equality tracking constraint, C_3 is formulated as an integral. In addition, equality constraints on energy are *point* constraints. Both of these items will have profound effects on the example trajectories to be generated.

Parameterization of the Controls

To approximate optimum system performance from the aforementioned structural model, a suitable parameterization of the controls, $\tau_{1,2}(t)$ via α, is necessary. For this study, the simplest case was chosen. Tabular values of α, at equal-spaced fixed times, t_i, for both joints were chosen as parameters, or

$$ \alpha_1(t_i), \quad \alpha_2(t_i), \quad i = 1, n \qquad 0 \le t_i \le t_f, $$

which results in $2n$ control parameters.

However, since the final time is changing due to minimization, the loss of control history definition would result if the times at which the control parameters are defined remain fixed in an absolute sense. To correct this, $\alpha_{1,2}$ were specified at equally-spaced, nondimensional *node* points, $\zeta_i = t_i/t_f$, where

$$ \alpha_1(\zeta_i), \quad \alpha_2(\zeta_i), \quad i = 1, n \qquad 0 \le \zeta_i \le 1, $$

This allows the torque histories to "stretch" naturally over the trajectory length. Using this modification, it is necessary to add t_f as a parameter also, resulting in $2n + 1$ control parameters to be found. Linear interpolation was used to compute $\alpha(\zeta)$ between the node values.

Numerical derivatives of the performance index, t_f, and the constraints, $C_j(t_f)$, provided to VF02AD are central finite-difference approximations. In computing these approximations, complete trajectories (or integrations of Eq.6) are computed using the current nominal t_f to produce perturbed $C_j(t_f)$ values. Since derivatives are computed over the current *fixed* t_f, the derivatives, $\partial t_f/\partial(\alpha_{1,2})_i = 0$, and only the derivatives, $\partial C_j(t_f)/\partial(\alpha_{1,2})_i \ne 0$. Obviously, both $t_f, C_j(t_f)$ gradients with respect to t_f, evaluated over the current nominal torque histories, are nonzero.

Results

The following finite-element structural model for the manipulator was used to produce the sample trajectories.

ITEM	LENGTH (m)	MASS (kg)	EI (newton-m^2)
joint-1 bracket	.0635	.545	10^5
link-1	.5040	.640	$10^2, 10^3$
1st joint-2 bracket + joint-2	.1070	5.415	10^5
2nd joint-2 bracket	.1040	.830	10^5
link 2	.4890	.313	$10^2, 10^3$
Totals:	1.2675	7.743	

Brackets were modeled with 1 element and considered rigid ($EI = 10^5$), and links were modeled with 3 elements for a total of 9 elements. The two values of stiffness, EI, for links 1,2 represent the trajectory comparison for this study. Point moments of inertia were used to define mass distribution for the brackets. No payload was used in this comparison. The joints were assumed to have no compliance or damping.

The two trajectories, computed on a CRAY-XMP, were integrated for 100 time steps, where $\Delta t = .01t_f$. Trajectory evaluations for gradient computations executed in 0.75 secs. The $\alpha(\zeta)$ histories for each joint were composed of 21 tabular values, where $\Delta\zeta = .05$. Torque bounds were chosen as $\pm16, \pm4$ newton-m (n-m) for joints 1 and 2. The path to be tracked for this study was the line connecting (x, y) pairs, $(0.0, 1.13)$ and $(1.13, 0.0)$.

Fig.2 shows the τ_1 profiles. These retain much of the bounded appearance of switching functions for purely rigid configurations. However, they begin and end near zero instead of the bounds (±16 n-m). These torque profiles exhibit

very similiar behavior, except at the intermediate switch point. Other than the slight ripples for the $EI_{link_s} = 100$ case, this region is the major difference between the configurations. Note the abruptness of the controls near the end in an attempt to quiet the structure. Also note the very slight change in t_f for the "softer" structure. The τ_2 torques showed minimal activity for most of the trajectory, except close to the end in order to accomplish the rest state. Again, very little "torquing" difference was noted for the order of magnitude change in link stiffness between configurations.

The straight-line tracking error in millimeters (mm) is shown in Fig.3. Both torque histories appear to limit the error to less than ±0.5 mm except near the end where the error momentarily "escapes" to about 2 mm. One drawback to the integral formulation is that it can relax tracking performance in isolated parts of the trajectory, yet yield a reasonably low residual (≈ 0) for $C_3(t_f)$. It may be necessary to add interior point constraints to decrease this error.

Strain energy is shown in Fig.4 and graphically depicts the difference between the semi-rigid and flexible links. The semi-rigid structure produces relatively little strain (however does contain a high frequency ripple), while the flexible-link configuration contains a 17 Hz mode with a sizable increase in energy magnitude. Note the abrupt changes in both cases, mirroring the sharp τ_1 changes in Fig.2. Also, note the enforcement of the $SE(t_f) = 0$ point constraint at the end.

Conclusions

A robust, parameter optimization tool has been successfully employed to generate actuator torque histories for approximate, minimum-time slewing maneuvers containing continuous and point constraints for a 2-link flexible manipulator. The parameters, or actuator torques, for each link were tabular values at fixed node points during the maneuver. Perturbations were made to each parameter to approximate final time and constraint gradients. The efficient formulation of the finite-element model made the numerical optimization procedure a realistic endeavor.

The accuracy of the straight-line tip tracking was excellent. For the trajectory used in this study, joint-1 applied most of the input in a manner resembling rigid-link torquing. Torque histories were very similiar between configurations even though link stiffness varied by an order of magnitude, and the t_f performance index was virtually the same. Energy and acceleration constraints were effective in bringing the structure nearly to rest at t_f. It was also demonstrated that final energy constraints do not preclude vibrations during the slew. The intended production use of the manipulator will dictate whether or not this is a hindrance.

References

[1] J.E. Bobrow, S. Dubowksy and J.S. Gibson, "Time-Optimal Control of Robotic Manipulators along Specified Paths", *Int. J. Robotics Res.*, Vol 4.,no.3, Fall 1985

[2] Weinreb, A., Bryson, A.E., "Optimal Control of Systems with Hard Control Bounds", *IEEE Transactions on Automatic Control*, Vol.AC-30, No.11, Nov. 1985, pp.1135-1138

[3] E.B. Meier and A.E. Bryson, "An efficient algorithm for time-optimal control of a two-link manipulator,", in AIAA conference on Guidance, Navigation, and Control, Monterey, CA, Aug 1987, pp 204-212.

[4] Bobrow, J.E., "Optimal Robot Path Planning Using the Minimum-Time Criterion", *IEEE J. Robotics and Automation*, Vol.4, No.4, August 1988

[5] Powell, M.J.D., "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations", *Proceedings of the Biennnial Conference on Numerical Analysis*, Springer-Verlag, Berlin, 1978, pp.144-157

[6] Eisler, G.R., Hull, D.G., "Maximum Terminal Velocity Turns at Nearly Constant Altitude", *Journal of Guidance, Control, and Dynamics*, Vol.11, No.2, March-April 1988, pp.131-136

[7] Outka, D.E., "Parameter Optimization Capability in the Trajectory Code PMAST", SAND86-2917, Sandia National Laboratories, Albuquerque, 1987

[8] Robinett, R.D., "A Unified Approach to Vehicle Design, Control, and Flight Path Optimization", The Center for Strategic Technology, Texas A&M University, SS87-1, 1987

[9] J. C. Simo and L. Vu-Quoc 'The role of non-linear theories in transient dynamic analysis of flexible structures', *Journal of Sound and Vibration, Vol. 119*, 1987, pp 487-508.

[10] T. R. Kane, R. R. Ryan, and A. K. Banerjee, 'Dynamics of a cantilever beam attached to a moving base', *Journal of Guidance, Control and Dynamics*, *Vol. 10*, 1987, pp 139-151.

[11] S. Hanagud and S. Sarkar, 'Problem of the dynamics of a cantilever beam attached to a moving base', *Journal of Guidance, Control and Dynamics*, Vol. 12, 1989, pp 438-441.

[12] A. E. Green and W. Zerna, *Theoretical Elasticity*, Clarendon Press, Oxford, 1954.

[13] D. J. Segalman, 'A Mathematical Formulation for the Rapid Simulation of a Flexible Multilink Manipulator' SAND89-2308, Sandia National Laboratories, Albuquerque, New Mexico
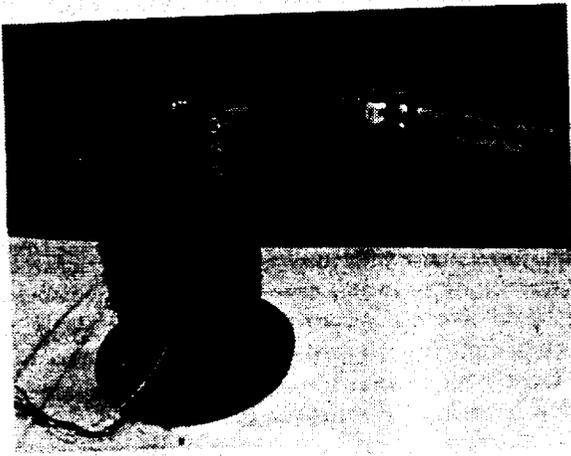
Figure 1: Sandia two-link manipulator
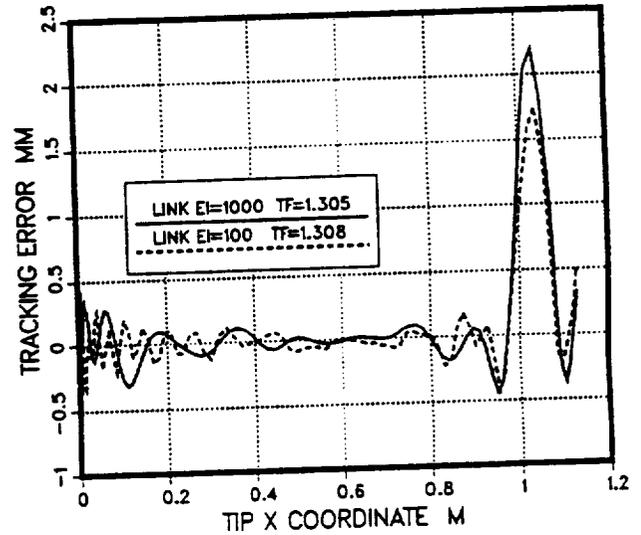
Figure 3: Straight-line tracking error
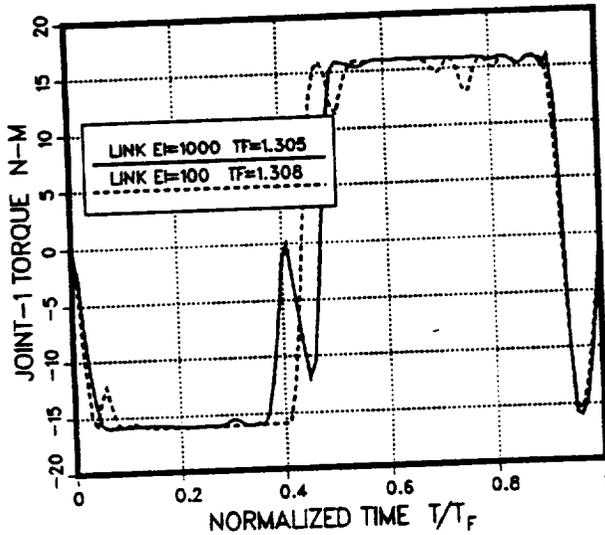
Figure 2: τ_1 joint torque histories
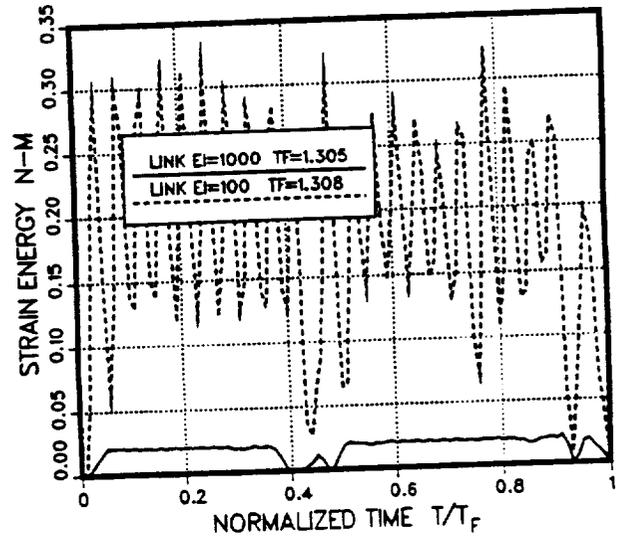
Figure 4: Strain energy histories

Improved Approximations of Displacements for Structural Optimization

Uri Kirsch

Department of Civil Engineering, Technion, Haifa 32000, Israel $S_{5}5-39$

1.INTRODUCTION

In most structural optimization problems the implicit behaviour constraints are evaluated for successive modifications in the design. For each trial design the analysis equations must be solved and the multiple repeated analyses usually involve extensive computational effort. This difficulty motivated several studies on explicit approximations of the structural behaviour in terms of the design variables[1-8]. The latter approach can considerably reduce the amount of computations, but the quality of the approximations might not be sufficient. Many of the approximate behaviour models proposed in the past are valid only for relatively small changes in the design variables. The aacuracy of the results is often insufficient for large changes in the design.

The object of this study is to present efficient and high quality approximations of the structural behaviour. It will be shown that the quality of the approximations can greatly be improved by combining scaling of the initial design, using intervening variables and scaling of a set of fictitious loads. Integrating these means, a powerful solution procedure can be introduced. In addition, the errors in satisfying the analysis equations can readily be evaluated. A numerical example illustrates the solution methodology and the effectiveness of the proposed approach.

2.PROBLEM FORMULATION

The problem under consideration can be stated as follows. Given an initial design X^*, the corresponding stiffness matrix K^* and the displacements r^*, computed by the equilibrium equations

$$K^* r^* = R \tag{1}$$

where R is the load vector, whose elements are assumed to be constant. Assume a change ΔX in the design variables so that the modified design is

$$X = X^* + \Delta X \tag{2}$$

and the corresponding stiffness matrix is

$$K = K^* + \Delta K \tag{3}$$

where ΔK is the change in the stiffness matrix due to the change ΔX. The object is to find efficient and high quality approximations of the modified displacements r due to various changes in the design variables X, without solving the modified equations

$$(K^* + \Delta K) r = R \tag{4}$$

__The scaling operation__ will be used in this study to improve the quality of the approximate displacements. Two types of scaling will be considered:

a) Scaling of the initial stiffness matrix K^* by

$$K = \mu K^* \tag{5}$$

where μ is a positive scalar multiplier. From Eqs. (1) through (5) it is clear that the precise displacements after scaling can be calculated directly by

$$r = \mu^{-1} r^* \tag{6}$$

b) Scaling of a ficticious load vector. Denoting any approximation of r by r_a, a corresponding fictitious load vector R_a can readily be calculated by Eq. (4)

$$R_a = (K^* + \Delta K) r_a \tag{7}$$

It can be noted that r_a are precise displacements for the stiffness matrix $K = K^* + \Delta K$ and the fictitious load vector R_a. Scaling of the loads R_a by

$$R_s = \Omega R_a \tag{8}$$

will give the precise displacements r_s due to R_s for the given stiffness matrix K, where

$$r_s = \Omega r_a \tag{9}$$

__Taylor series expansion__ is one of the most commonly used approximations in structural optimization. The first order approximations of r about X^* are given by

$$r_a = r^* + r_x^* (X - X^*) \tag{10}$$

where both r^* and r_x^* are computed at X^*. The matrix of first derivatives r_x^* can be computed by several methods. A major problem in using Eq. (10) is the accuracy of the results. The discrepancy in satisfying the equilibrium equations (4) due to the approximate displacements is given by (Eq. (7))

$$\Delta R = R_a - R = (K^* + \Delta K) \, r_a - R \tag{11}$$

Evidently, $\Delta R = 0$ for the precise displacements. Thus, ΔR can be used to evaluate the quality of the approximations. This criterion, combined with the two types of scaling discussed earlier will be used now to introduce improved approximations.

3. IMPROVED APPROXIMATIONS BY SCALING

Scaling of the initial stiffness matrix may improve the quality of the approximations, if the known modified displacements (Eq. (6)) provide better initial data then the original displacements. The approach proposed here is suitable for various types of design variables (such as geometrical variables). For illustrative purposes it is assumed that the displacement are homogeneous functions of X.

Homogeneous functions of degree n in the design variables are defined by

$$r(\mu X^*) = \mu^n \, r(X^*) \tag{12}$$

where μ is a positive scalar. Euler's theorem on homogeneous functions states that

$$r^*_X \, X^* = n \, r^* \tag{13}$$

The derivatives of homogeneous functions of degree n are given by

$$r_X(\mu X^*) = \mu^{n-1} r_X(X^*) \tag{14}$$

These properties of homogeneous functions can be used to obtain simplified approximations [2]. Substituting Eq. (13) into Eq. (10) yields

$$r_a = (1-n) r^* + r_X^* \, X \tag{15}$$

Assume a point X along the scaling line

$$X = \mu X^* \tag{16}$$

Expanding Eq. (15) about μX^*, we have by substituting Eqs. (12) and (14),

$$r_a = (1-n) \, \mu^n \, r^* + \mu^{n-1} \, r_X^* \, X \tag{17}$$

It can be noted that precise solutions are obtained by Eq. (17) along the scaling line (16). The multiplier μ can be selected such that the approximations are improved. Further improvements could be achieved by assuming intervening variables.

Intervening variables. Assume intervening variables of the form

$$Y_i = X_i^m \tag{18}$$

The displacements r in this case are homogeneous functions of degree N in Y, where $N = n/m$. Therefore, the first order Taylor series expansion (Eq. (15)) is

$$r_a = (1 - n/m) \, r^* + r_y^* \, Y \tag{19}$$

This equation can be expressed in terms of X. Substituting Eq. (18) and

$$r_y^* = m^{-1} \, X^{*1-m} \, r_X^* \tag{20}$$

into Eq. (19) yields

$$r_a = (1 - n/m) \, r^* + m^{-1} \, r_X^* \, X^{*1-m} \, X^m \tag{21}$$

Expanding the series about μX^*, this expression becomes (see Eqs. (12) and (14))

$$r_a = \mu^n (1 - n/m) r^* + \mu^{n-1} \, m^{-1} r_X^* \, X^{*1-m} \, X^m \tag{22}$$

Special cases. For any given n, the values of μ and m can be chosen to improve the approximations. The following special cases of Eq. (22) might be considered:

-*No scaling or intervening variables* $\mu = m = 1$ *(Eq. (15)).*
-*No intervening variables, only scaling* $m=1$ *(Eq. (17)).*
-*No scaling, only intervening variables* $\mu = 1$ *(Eq. (21)).*
-*The usual cross-sectional variables, only scaling* $m=1, n=-1$

$$r_a = 2 \, \mu^{-1} r^* + \mu^{-2} \, r_X^* \, X \tag{23}$$

-*The inverse variables with scaling* $m = n = -1$

$$r_a = - \mu^{-2} \, r_X^* \, X^{*2} \, X^{-1} = \mu^{-2} r_y^* \, Y \tag{24}$$

-*The inverse variables, no scaling* $m = n = -1, \mu = 1$

$$r_a = r_y^* \, Y \tag{25}$$

Since $r_y(\mu Y^*) = r_y^*$ (Eq. (14)), the approximations (25) are precise along the scaling line $Y = \mu \, Y^*$. This illustrates the advantage of using the inverse variables for approximations near this line in truss

structures with cross-sectional areas as variables.

Design procedure. The multiplier μ can initially be selected based on geometrical considerations. Assuming for example the criterion

$$\mu = (\Sigma X_i^2 / \Sigma X*_i^2)^{0.5} \tag{26}$$

then the distance between X and the origin of the design space and the distance between $\mu X*$ and that origin will be identical. A major drawback of this approach is that the nature of the structural behaviour is not taken into consideration. Another criterion, proposed here, is to minimize the errors in the approximate displacements. Evaluating r_a by Eq. (22), the resulting R_a can readily be calculated by (7). The latter fictitious loads can then be scaled by Eq. (8) such that the final displacements Ωr_a (Eq. (9)) are improved. The discrepancy in satisfying the equilibrium conditions is given by (Eq. (11))

$$\Delta R(\Omega) = \Omega R_a - R \tag{27}$$

Defining the common measure of smallness of $\Delta R(\Omega)$ by the quadratic form

$$Q(\Omega) = (\Omega R_a - R)^T (\Omega R_a - R) \tag{28}$$

the optimal Ω value is selected such that $Q(\Omega)$ is minimized. Differentiation of Q with respect to Ω and setting the result equal to zero yields

$$\Omega = (R_a^T R) / (R_a^T R_a) \tag{29}$$

Based on these considerations, the following solution procedure is proposed:

a) Initial values for μ (Eq. (26)) and m are assumed.

b) The approximate displacements r_a are evaluated (Eq. (22)).

c) The values of R_a (Eq. (7)) Ω (Eq. (29)) and Q (Eq. (28)) are calculated.

d) The value of Q is checked for optimality; μ and m are then modified and steps b,c are repeated as necessary.

e) The final approximate displacements are evaluated by Eq. (9).

The solution methodology and the effectiveness of the proposed procedure are illustrated by the numerical example.

4. NUMERICAL EXAMPLE

Consider the truss shown in Fig. 1 with four cross-sectional area design variables Xi (i=1...4). Assume the initial design $X = 1.0$ with the given displacements $r*^T = \{1.818, 1.604, 1.107\}$ (the modulus of elasticity is 30,000).

The effectiveness of the inverse variables formulation (Eq. (25)) is demonstrated in Table 1 for various changes in the design near the initial scaling line. It can be noted that excellent results have been obtained for changes in the variables up to +900% (cases a-d) and -91% (cases e-h).

To illustrate the effect of m and μ on the results, assume now the modified design $X^T = \{5.0, 10.0, 10.0, 10.0\}$, where the change in the design variables is up to 900%. That is, the magnitude of ΔX is relatively large. The precise solution and results obtained by Taylor series (Eq. (15)), Initial scaled design (Eqs. (23) and (26)), and Inverse variables formulation (Eq. (25)) are summarized in Table 2. The first order Taylor series expansion provides meaningless results. Better, yet insufficient results are obtained by the initial scaled design and the inverse variables formulation.

The effectiveness of the proposed solution procedure is illustrated for two cases:

a) Optimization of μ with no intervening variables (Eq. (23), see Table 3).

b) Optimization of m with no scaling of the initial design (Eq. (21), see Table 4).

Precise results have been obtained in both cases for the optimal μ and m values. However, the results are sensitive to changes in these parameters near the optimum.

5. CONCLUSIONS

Approximations of the structural behaviour in terms of the design variables are essential in optimization of large structures, where the time consuming analysis must be repeated many times. A major problem is that the quality of the commonly used approximations might not be sufficient, particularly for large changes in the design.

A general solution procedure to obtain effective approximations is proposed in this study. Scaling of the initial stiffness matrix, intervening variables and scaling of fictitious loads are combined to achieve efficient and high quality approximations. Using the proposed approach, the errors in satisfying the

equilibrium equations due to the approximate displacements can readily be evaluated without performing an exact analysis. A simple example illustrates the solution methodology and the effectiveness of the presented procedure. It is shown that very good results have been obtained for large changes in the design variables. The solution procedure is suitable for various types of design variables. For illustrative purposes, homogeneous displacement functions have been assumed. The following observations have been made:

-The type of intervening variables can be selected for homogeneous functions. Two disjoint (positive and negative) feasible regions might be obtained for m ($m \neq 0$). The approximations might be sensitive to changes in m; that is, the optimal m can significantly improve the results.

-Scaling of the initial stiffness matrix is most effective in improving the quality of the approximations. Very good results have been obtained with a relatively small computational effort. The results might be sensitive to changes in the value of the scaling multiplier and it is therefore advantageous to optimize μ.

-The results can further be improved by scaling of the fictitious loads. It has been shown that significant improvements have been achieved even in cases where scaling of the initial stiffness matrix provided poor results. This type of scaling can be applied such that the errors in satisfying certain equilibrium equations will be reduced.

In summary, the proposed solution procedure is a powerful tool to achieve efficient and high quality approximations. it also provides insight and better understanding of structural behaviour models.

6. REFERENCES

1. Abu Kassim, A.M. and Topping, B.H.V. (1987) Static reanalysis of structures:A review, J. Struct. Eng. ASCE, 113, 1029-1045.
2. Hjali, R.M. and Fuchs, M.B. (1989) Generalized approximations of homogeneous constraints in optimal structural design, in Computer aided optimum design of structures (Eds. C.A. Brebbia and S. Hernandez), Springer-Verlag, Berlin.
3 .Kirsch, U. and Toledano, G. (1983) Approximate reanalysis for modifications of structural geometry, Computers and structures, 16, 269-279.
4. Kirsch, U. (1984) Approximate behaviour models for optimum structural design, in New directions in optimum structural design (Eds. E. Atrek, et al),John Wiley &Sons, NY.
5. Kirsch, U. and Taye, S. (1988) High quality approximations of forces for optimum structural design, Computers and structures, 30, 519-537.
6. Kirsch, U. (1989) Approximate models for structural optimization, presented in NATO ASI on Optimization and decision support systems in civil engineering, Edinburgh, U.K.
7. Prasad, B (1983) Potential forms of explicit constraint approximations instructural optimization, Comp. meth. appl. mech. engrg. 241-261.
8. Schmit, L. A. and Farshi, B. (1974) Some approximation concepts for structural synthesis, AIAA J. 11, 489-494.

Figure 1: Seven-bar truss

Figure 1: Seven-bar truss
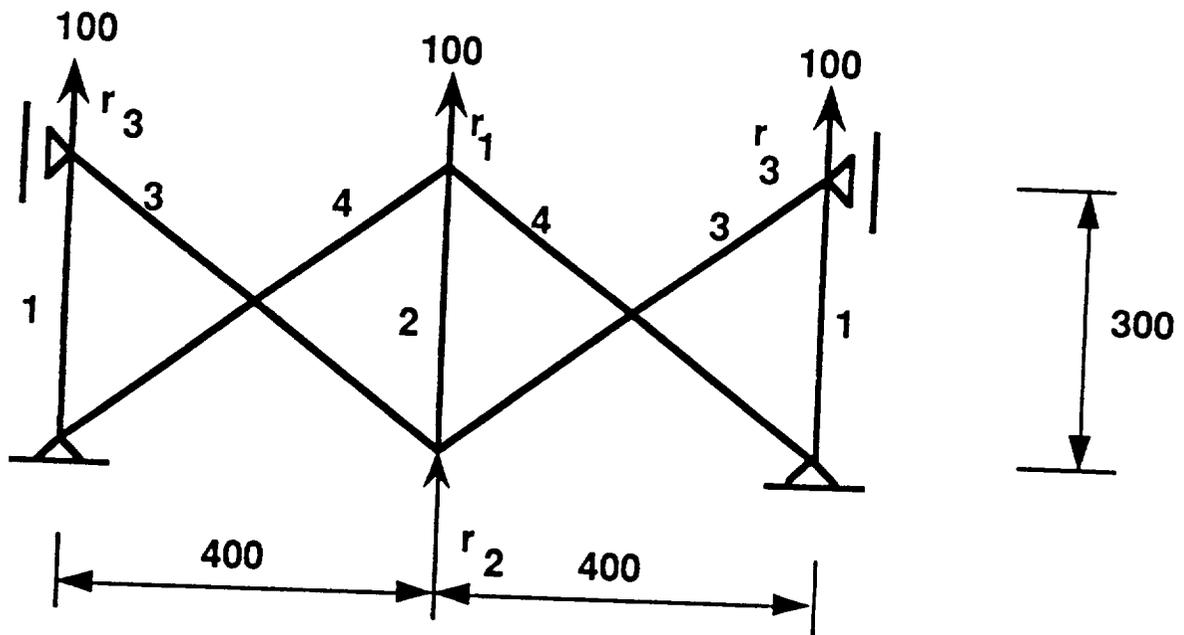
Table 1: Results, seven-bar truss, Eq. (25)

Case	X^T				$30\,r_a$			R_a^T		
a	9.09	10.0	10.0	10.0	5.58	4.99	3.63	100.00	0.00	100.20
b	10.0	9.09	10.0	10.0	5.48	4.78	3.32	100.03	0.00	100.00
c	10.0	10.0	9.09	10.0	5.51	4.64	3.31	100.00	0.00	99.96
d	10.0	10.0	10.0	9.09	5.79	5.06	3.37	100.26	0.00	100.00
e	0.09	10.0	10.0	10.0	558	499	363	100.00	0.00	100.20
f	10.0	0.09	10.0	10.0	548	478	332	100.03	0.00	100.00
g	10.0	10.0	0.09	10.0	551	464	331	100.00	0.00	99.96
h	10.0	10.0	10.0	0.09	579	506	337	100.26	0.00	100.00

Table 2: Results, seven-bar truss $X^T = \{5, 10, 10, 10\}$

Method	r_a^T			R_a^T		
Precise	0.221	0.217	0.206	100.00	0.00	100.00
Eq. (15)	-12.5	-9.9	-3.8	-	-	-
Eq. (23)	0.224	0.221	0.212	100.00	0.00	104.52
Eq. (25)	0.206	0.196	0.172	98.80	0.00	80.95

Table 3: Seven-bar truss, effect of μ and Ω, Eq. (23)

μ	r_a^T			Ω	$\Omega\,R_a^T$		
7.5	88.88	0.00	92.90	1.100	97.80	0.00	102.20
7.7	91.10	0.00	91.10	1.098	100.00	0.00	100.00
8.0	93.74	0.00	88.63	1.097	102.80	0.00	97.20

Table 4: Seven-bar truss, effect of m and Ω, Eq. (21)

m	R_a^T			Ω	$\Omega\,R_a^T$		
-1.00	100.00	0.00	104.52	0.978	97.80	0.00	102.20
-1.01	106.50	0.00	106.50	0.939	100.0	0.00	100.00
-1.02	113.22	0.00	108.18	0.903	102.30	0.00	97.70

Parallel Structural Optimization with Different Parallel Analysis Interfaces

Mohamed E. M. El-Sayed and Ching-Kuo Hsiung
Department of Mechanical and Aerospace Engineering University of Missouri-Columbia, Missouri 65211

ABSTRACT

The real benefit of structural optimization techniques is in the application of these techniques to large structures such as full vehicle or full aircraft. For these structures, however, the sequential computer's time and memory requirements prohibits the solutions. With the recent existence and rapid development of parallel computers, parallel processing of of large scale structural optimization problems is achievable. In this paper we discuss the parallel processing of structural optimization problems with parallel structural analysis. Two different types of interface between the optimization and Analysis routines are developed and tested.

1. INTRODUCTION

For the large-scale structural design problems, repeated finite element calculations consume a lot of CPU time and makes the optimum design procedure slow and intractable. Because of this, various structural reanalysis techniques have been developed. State-of-the-art reviews of reanalysis techniques can be found in [1-3].

The fifth generation computers, which were introduced in the 80s, have several advantages over the fourth generation machines: higher speed of computation, bigger central memory, and a multi-processor structure. All of these advantages allow the researchers to move further in the structural optimization area. Venkayya [4],vectorized some finite element programs and utilized the vector processor and huge memory of the Cray-1 to get an improvement in computational efficiency of almost two orders of magnitude. In addition, various parallel finite element methods have been developed; however, none were applied to the structural optimization area.

Since all optimization methods require design sensitivity analysis, it plays an important role in optimization procedures. Due to the repeated finite element analysis, the design sensitivity analysis becomes one of the most time-consuming parts of structural optimization. Although there are many papers addressing design sensitivity [5,6], articles describing design sensitivity analysis using parallel computers could not be found.

Since the amount of time spent in evaluating the gradient of the constraints in the optimization problem is computationally expensive and the computation of the gradient at each iteration involves a number of uncoupled calculations, Sikiotis and Saouma [7] spread the job of constraint gradient calculation to four Apollo workstations to reduce the calculation burden of one machine. This resulted in a relatively low overhead with an achieved speedup dependent on the size and nature of the problem, and the system configuration. The speedup increased with the problem size.

The optimum design of modern structures usually involves a large number of variables and constraints. While mathematical programming techniques provide the designers with the tools for optimization, these techniques can not handle large design problems. To overcome this weakness, the decomposition method was introduced [8,9]. The method depends on decomposing the original problem into a number of smaller subproblems and solving each subproblem separately. Since the subproblems after decomposition are coupled, iterative calculations become unavoidable [10,11].

Some of the most applied decomposition methods are the multi-level optimization approaches. In a multilevel optimization approach, the original design problem is decomposed into a top level design problem and some uncoupled subproblems in the other levels. The top level is concerned with the overall optimization problem while the detail design problems are handled by the lower levels. The optimization results are obtained by iterating between the different levels.

Different approaches have been proposed for multilevel optimization. For example, based on the state space formulation of the optimum control technique, Govil *et al.* presented an algorithm for structural optimization by substructuring [12-14]. Kirsch used the model and goal coordination methods [15] in the multilevel structural optimization [16-19]. Sobieszczanski-Sobieski *et al.* [20] proposed a general multilevel optimization method which not only considers the structural design but also takes into account other different disciplines, such as aerodynamics. Recently, other structural optimization methods by multilevel decomposition have been also proposed [21-23]. In [22], a three-level structural optimization procedure is illustrated by a portal frame. The portal frame is decomposed into several beams, and each beam is decomposed into plates. The bottom level in the multi-level optimization handles the plates, the intermediate level handles the beams and the top level handles the assembled structure.

This work represents a study for the use of parallel structural analysis method in solving structural optimization problems with two different types of interface between the optimization and Analysis routines.

2. FORMULATION

For structural optimization problems, the structural analysis is recognized as one of the most time-consuming parts. In the following, the structural analysis using a parallel structural analysis in the structural optimization problems is discussed.

A general optimum structural design problem may be defined as follows: find a design variable vector X subjected to constraints related to the design and the behavior of the structure that minimize an objective function F(X). The mathematical form could be expressed as:

Minimize: $F(X)$; $X = [x_1, x_2, ..., x_N]^T$ $X \in R^N$

Subject to:

$$A \leq X \leq B$$
$$\Phi_k (X) \geq 0 \qquad k=1,2,...,K$$
$$\Psi_l (X) = 0 \qquad l=1,2,...,L$$

Where

$F(X)$:	The objective function, usually the weight of the structure.
X :	The design variable vector.
N :	Total number of design variables.
$\Phi_k(X)$:	K inequality constraints function.
$\Psi_l(X)$:	L equality constraints function.
A :	The vector of lower bound on design variables.
B :	The vector of upper bound on design variables.

The constraints might be placed on the stress or displacement under the design loads. These constraints may not be expressed explicitly, but can be numerically evaluated using finite element analysis.

In the optimization calculations, repeated finite element analyses have to be performed in order to evaluate the behavior of the structure. To reduce the computational CPU time of the structural optimization process, the parallel finite element computation technique with separate substructures developed in [24] is adapted with a minor modification.

The main characteristic of the parallel finite element analysis technique developed in [24] is that the finite element analysis of each substructure can be processed independently with limited communications. The basic idea of the structural optimization algorithm developed here is that whenever it is necessary to use the finite element analysis, the calculation loads will be spread to every assigned processor. Each processor, including the main processor and associate processors, will handle the structural analysis calculation of its own assigned substructure.

3. ALGORITHM

In this approach, one processor, called main processor, is chosen to execute the optimization calculations and the finite element analysis of one substructure. The other associate processors are used to analyze the assigned substructures. The major steps for the main processor are:

Step 1: Start the optimization calculations. When there is a need to do the structural analysis, the main processor will start the parallel finite element analysis software. The related design variables will be sent to the corresponding associate processors.

Step 2: Perform the finite element analysis of the assigned substructure and assemble the stiffness and force contribution matrices from the other processors.

Step 3: Solve the displacements at the shared boundary nodes and send these data to the corresponding associate processors.

Step 4: Continue the calculation of constraint-related data of the assigned substructure, such as stress and displacement, and receive the constraint-related data from the associate processors.

Step 5: After the constraint-related data of the complete structure are collected, check the convergence. If convergence is obtained then stop, otherwise go to Step 1.

The major steps for the associate processors are:

Step 1 : Receive the corresponding design variables from the main processor, start the finite element computations on the substructure assigned, then send the contributed stiffness and force data on the common boundary back to the main processor.

Step 2 : Receive the boundary displacements from the main processor, continue the finite element computations to solve the domain displacement and stress and send the constraint-related data back to the main processor.

At the end of Step 2, the associate processors are kept idle until the next finite element computation.

4. INTERFACING METHODS BETWEEN ANALYSIS AND OPTIMIZATION ROUTINES

The previous algorithm offers a straightforward method to distribute the finite element computational loads in the structural optimization process. Two different kinds of interface between optimization software and finite element software can be developed.

Design software is a combination of optimization software and analysis software, as shown in Figure 1. They are usually independent of each other. In the optimum structural design area, reliable and powerful analysis software such as NASTRAN and ANSYS are often complicated. The interface problem between optimization software and analysis software, therefore, becomes important [25]. In this work, two different kinds of interface between optimization software and finite element software have been developed and tested.

The first Method is a multi-reading interface. In this method, the optimization software and the finite element analysis software communicate via the named pipes of NP-routine [26] and the data file. Instead of creating an input file for finite element analysis software repeatedly, the optimization software transfers the new design variables to the finite element software directly. The rest of the data needed for the finite element analysis software were read from the original data file. The output data from the finite element analysis software is also transferred to the optimization software via the named pipes.

The second method is a single-reading interface. In this method, the communication between the optimization software and the finite element software remain the same as the multi-reading interface. Instead of reading the data from the original data file repeatedly, however, work has been done to extract the input data and store them in core memory after the first time reading. Therefore, the finite element software will read the input data file just one time.

5. TEST CASES

The test cases were performed on the Cray X-MP four-processor supercomputer. The results obtained are compared with the sequential version of the code on the same machine. The input data and conditions on both parallel and sequential version of the code were the same.

Figure 1 shows the 200-member truss structure. In these test cases, the structure is divided into two, three and four substructures.

Two Substructures Case:
The truss is divided into two substructures by partitioning it at joints 34 to 42 and 57 to 61. Members 1 to 105, 123 to 127, 140 to 143, 152 and 153 are in substructure 1. The rest of the members are in substructure 2.

Three Substructures Case:
The truss is divided into three substructures by partitioning it at joints 34 to 36, 20 to 28, 50 to 59, and 42 to 44. members 1 to 50, 59 to 63, and 81 to 83 are in substructure 1. Members 140 to 200 and 123 to 127 are in substructure 3. The rest of the members are in substructure 2.

Four Substructures Case:
The truss is divided into four substructures by partitioning it in the x1 direction at joints 3, 5, 7, 9. After the partitioning the structure has five levels. The bottom level belongs to substructure 4. The second level and the fourth level belong to substructure 1. The third level belongs to substructure 3. The top level belongs to substructure 1.

The objective function is the total weight of the structure. The design variables are the cross-sectional areas of the truss and can be reduced using simple design variable linking. In the first test case, the members 1 to 105 and members 106 to 200 are linked to design variable number 1 and number 2, respectively. Stress constraints were placed on every member of the truss. The design data for this structure is given in Table 1.

Table 2 contains the computational time and speedup values for the 200-member problem using multi-reading interface with two processors. Table 3 represents the computational time and speedup values for the same problem using single-reading interface with two processors.

Additional test cases were performed using single reading interface. In these test cases the structure was divided into 20 sub-groups by partitioning it in the x1-direction at joints 3, 5, 7, 9 and in the x2-direction at joints 28, 44, 61. The members of each substructure were then linked as a design variable. The computational time and speedup values, for these test cases with different number of processors, are given in Table 4.

Table 1 The design data for 200-member truss structure.

Modulus of elasticity	3.0×10^4 ksi
Density of material	0.283 lb/in^3
Member-size constraints:	$0.1 \leq X_i$ i=1,200
Stress constraints:	-30.0 ksi $\leq \sigma \leq 30.0$ ksi
Loading condition:	one kip acting in the positive x1 direction at nodal points 1 to 11.

Table 2 Performance of 200-member truss structure
Two substructures, 2 design variables
By multi-reading interface

	CPU time (sec.)	Speedup
Sequential calculation	6.239	
Parallel calculation	4.116	1.52

Table 3 Performance of 200-member truss structure
Two substructures, 2 design variables
By single-reading interface

	CPU time (sec.)	Speedup
Sequential calculation	3.319	
Parallel calculation	2.965	1.13

Table 4 Performance of 200-member truss structure
Two, three and four substructures, 20 design variables
By single-reading interface

	CPU time (sec.)	Speedup
Sequential calculation	13.518	
Parallel calculation		
2 processors	11.964	1.13
3 processors	11.286	1.20
4 processors	10.585	1.28
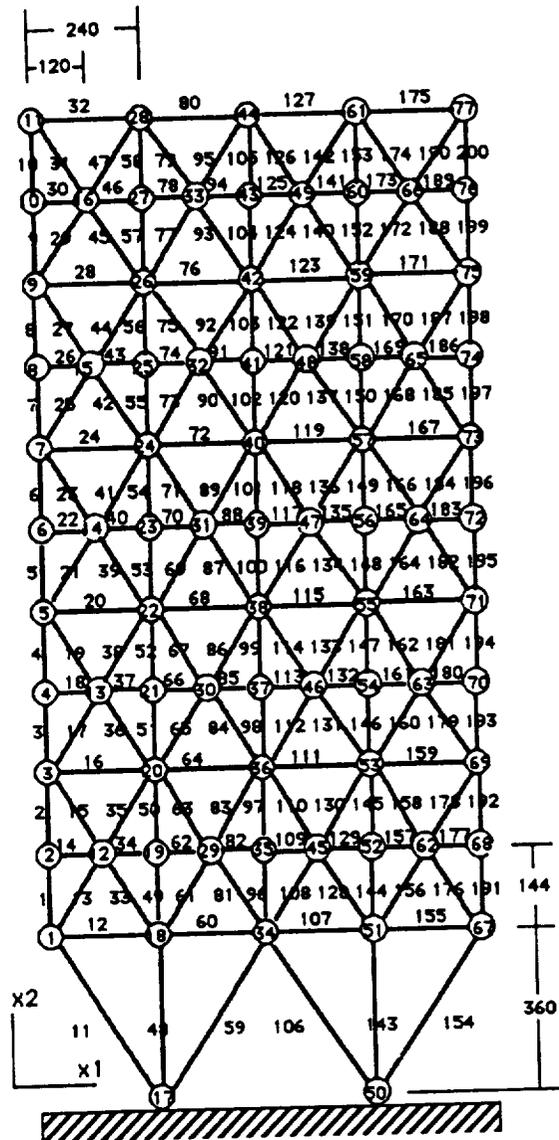
Figure 1 200-Member truss

6. DISCUSSION AND CONCLUSION

A structural optimization method with parallel structural analysis has been developed. In this method, a parallel finite element computation technique with separate substructures is adapted and the original structure was decomposed into several substructures. One processor on the Cray X-MP system was chosen to execute the optimization calculation and the finite element analysis of one substructure. The other processors then perform the structural analysis on the assigned substructures.

Two methods are also tested for interfacing the finite element analysis software with the optimization software. The test cases performed show that the speedup values increase as the number of processors increase. The single-reading interface method shows an advantage over the multi-reading interface method in the CPU time needed for calculation. Even though the speedup values of the multi-reading interface case are higher than the speedup values of the single-reading interface case, the saving in CPU time shows that the single reading interface is recommended.

402

7. REFERENCES

1. Arora, J. S., "Survey of structural reanalysis techniques", Journal of Structural Division, ASCE, No. 102, pp 783-802, 1976.

2. Abu Kassim, A. M., and Topping, B. H. V., "Static reanalysis: a review", Journal of Structural Division, ASCE, No. 113, pp 1029-1045.

3. Kitis, L., Pilkey, W. D., and Hirai, I., "Reanalysis", in Kardestuncer, H., and Norrie, D. H. (editors), Finite Element Handbook, McGraw-Hill, New York, 1987, Chapter 2, Part 4, pp 4.109-4.120.

4. Venkayya, V. B., Calahan, D. A., Summers, P. M., and Tischler, V. A., "Structural optimization on vector processors", Reference 8, pp 155-170, ASME, 1983.

5. Arora, J. S. and Haug, E. J., "Methods of Design Sensitivity Analysis in Structural Optimization", AIAA Journal, Vol 17, No. 9, pp 970-974, September 1979.

6. Haug, E. J., Komkov, V. and Choi, K. K., "Design Sensitivity Analysis of Structural Systems", Academic Press, 1984.

7. Sikiotis, E. S., and Saouma, V. E., "Parallel structural optimization on a network of computer workstations", Computers & Structures, Vol. 29, No. 1, pp 141-150, 1988.

8. Giles, G. L. "Procedure for automating aircraft wing structural design", Journal of the Structural Division, ASCE, Vol. 97, No. ST1, pp 97-113, January 1971.

9. Kirsch, U., Reiss, M., and Shamir U. " Optimum design by partitioning into substructures", Journal of the Structural Division, ASCE, Vol. 98, No. ST1, pp 249-267, January 1972.

10. Sobieszczanski, J. and Loendorf, D., "A mixed optimization method for automated design of fuselage structures", Journal of Aircraft, Vol. 9, No. 12, pp 805-811, December 1972.

11. Hughes, O., "A method for nonlinear optimum design of large structures and applications to naval ship design", International Symposium on Optimum Structural Design, Tucson, Arizon, pp 5-10, October 1981.

12. Arora, J. S. and Govil, A. K., "An efficient method for optimal structural design by substructuring", Computers & Structures, Vol. 7, pp 507-515, 1977.

13. Govil, A. K., Arora, J. S. and Haug, E. J., "Optimal design of wing structures with substructuring", Computers & Structures, Vol. 10, pp 899-910, 1979.

14. Govil, A. K., Arora, J. S. and Haug, E. J., "Optimal design of frames with substructuring", Computers & Structures, Vol. 12, pp 1-10, 1980.

15. Mesarovic, M. D., Macko, D., and Takahara, Y., "Two coordination principles and their application in large scale system control", Proceeding IFAC Congress, Warsaw, Poland, 1969.

16. Kirsch, U., "Optimum structural design-concepts, methods and applications. McGraw-Hill, New York, 1981.

17. Kirsch, U., "Multilevel optimal design of reinforced concrete structures", Engineering Optimization, Vol. 6, pp 207-212, 1983.

18. Kirsch, U., "Multilevel synthesis of standard building structures", Engineering Optimization, Vol. 7, pp 105-120, 1984.

19. Kirsch, U., "A improved multilevel structural synthesis method", Journal of Structural Mechanics, Vol. 13, pp 123-144, 1985.

20. Sobieszczanski-Sobieski, J., Barthelemy, J.-F. M. and Giles, G. L. "Aerospace engineering design by systematic decomposition and multilevel optimization", ICASE, pp 828-840, Toulouse, France, 1984.

21. Sobieszczanski-Sobieski, J., James, B. B. and Dovi, A. R., "Structural optimization by multilevel decomposition. AIAA Journal, Vol. 23, No. 11, pp 1775-1782, 1985.

22. Sobieszczanski-Sobieski, J., James, B. B. and Riley, M. F., "Structural optimization by generalized, multilevel optimization", Proceeding AIAA/ASME/ASCE/AHS 26th Structures, Structural Dynamics and Material Conference, pp 349-364, Orlando, Florida, 1985.

23. Sobieszczanski-Sobieski, J., "Multilevel structural optimization", NATO/NASA/NSF/USAF Advanced Study Institute on Computer Aided Optimal Design, Toria, Portugal, Vol. 3, pp 7-28, 1986.

24. El-Sayed, M. E. and Hsiung C. K., " Parallel Finite Element Computation with Separate Substructures", to be published by Computers & Structures.

25. Balling, R. J., Parkinson, A. R., and Free, J. C., "Method for interfacing analysis software to optimization software", Computers & Structures, Vol. 22, No. 1, pp 87-98, 1986.

26. Cray Research, Inc. Named-Piped Routine Manual, Mendota Heights, 1988.

RECENT EXPERIENCE WITH MULTIDISCIPLINARY ANALYSIS AND OPTIMIZATION IN ADVANCED AIRCRAFT DESIGN

by

Samuel M. Dollyhigh and
Jaroslaw Sobieszczanski-Sobieski
NASA Langley Research Center

INTRODUCTION

The task of modern aircraft design has always been complicated due to the number of intertwined technical factors from the various engineering disciplines. Furthermore, this complexity has been rapidly increasing by the development of such technologies as aeroelasticity tailored materials and structures, active control systems, integrated propulsion/airframe controls, thrust vectoring, and so on. Successful designs that achieve maximum advantage from these new technologies require a thorough understanding of the physical phenomena and the interactions among these phenomena. A study commissioned by the Aeronautical Sciences and Evaluation Board of the National Research Council has gone so far as to identify technology integration as a new discipline from which many future aeronautical advancements will arise (reference 1). Regardless of whether one considers integration as a new discipline or not, it is clear to all engineers involved in aircraft design and analysis that better methods are required. In the past, designers conducted parametric studies in which a relatively small number of principal characteristics were varied to determine the effect on design requirements which were themselves often diverse and contradictory. Once a design was chosen, it then passed through the various engineers' disciplines whose principal task was to make the chosen design workable. Working in a limited design space, the discipline expert sometimes improved the concept, but more often than not, the result was in the form of a penalty to make the original concept workable. If an insurmountable problem was encountered, the process began over. Most design systems that attempt to account for disciplinary interactions have large empirical elements and reliance on past experience is a poor guide in obtaining maximum utilizations of new technologies. Further compounding the difficulty of design is that as the aeronautical sciences have matured, the discipline specialist's area of research has generally narrowed as more sophisticated methods are developed in the specialist's area of expertise. The results have been a decrease in the awareness of the impact of his decisions on other disciplines.

On the other hand, advances in computer technology have greatly increased the capability to solve complex problems, display and manipulate data in more suitable engineering formats, and computers have become much more user friendly and easier to interact with. The modern workstation in particular enables the engineer to run several computer programs simultaneously, conveniently display results, interactively modify data, and, in general, efficiently proceed through a series of calculations. Workstations provide a dramatic increase in the capability of an airplane designer to generate and modify numerical models of the vehicle--a capability necessary for advanced aircraft design.

After assessing the environment just described, NASA Langley Research Center made a commitment to improved multidisciplinary research at the Center. A high-level Multidisciplinary Research Advisory Committee was formed and subsequently the High-Speed Airframe Integration Research (HiSAIR) project was initiated. HiSAIR focuses on the High-Speed Civil Transport (HSCT) design activity as a case to foster the development of methodology to improve multidisciplinary analysis, design, and optimization of aircraft systems and to develop a computational environment favorable to such an activity. This paper will outline the progress and problems encountered in the analysis, design, optimization sensitivity analysis, mathematical modeling, and configurations control and the means by

which they are being solved. The breadth versus depth dilemma in analysis and design and the means for coping with that dilemma will be discussed. Finally, the all-important human aspects and the need for a new "culture" for doing business in an integrated, multidisciplinary design environment are discussed.

ORGANIZATION AND STRATEGY

NASA's Langley Research Center is an institution in which research and technology development in discipline research is the principal product on the aeronautical side of the Center. The Center is organized along discipline lines in order the successfully accomplish this objective. Langley does not manufacture aircraft so the role of multidisciplinary research is somewhat different than that in much of the remainder of the aeronautical community. The major benefits of vehicle integration studies in a research institution are technology evaluation and identification of high-payoff technologies which serves to help advocate for programs and allocate resources. An equally important role is to identify the probable application of a technology when its interaction with other disciplines are considered. The second role aids in developing the proper data bases to expedite the utilization of technologies by industry. The HiSAIR project was organized to strengthen Langley's ability to conduct such multidisciplinary vehicle design, analysis, and optimization studies.

The organization of HiSAIR is shown in figure 1. The project has no formal line management authority, and is conducted on somewhat of an ad hoc basis at the working level. The Advanced Vehicles Division and the Interdisciplinary Research Office share the responsibility for pulling the activity together. These two organizations have functional responsibilities to conduct vehicle systems studies and improved multidisciplinary optimization methods, but are located in different discipline areas-- aerodynamics and structures. A steering committee of division chiefs oversees the direction and scheduling of the research. There is a loose tie with the High-Speed Research (HSR) Program to provide a vehicle focus for the research. The HSR Program coordinates research on the High-Speed Civil Transport. The vehicle focus for HiSAIR is considered necessary so that methods are developed to solve actual vehicle design problems. Experience tends to indicate that multidisciplinary design systems are best developed to solve real problems, then generalized to encompass other vehicle types. There is also an informal tie to Lewis Research Center as a source of propulsion system data. This informal management works because of the commitment of Langley's senior management to improving multidisciplinary design and optimization.

The strategy for HiSAIR is shown in figure 2. Because the discipline research emphasis is at Langley, the interfaces between disciplines tends to lag advancements in analytical methods and computer equipment. The first step is to strengthen these interfaces, especially in geometry methods and data transfer to form a rapid high fidelity analysis system. The second step is to conduct a high fidelity (higher order methods) limited optimization using sensitivity analysis methods that are the subject of papers in this symposium (references 4 and 5). The longer term goal is to develop a full system optimization capability with optimization methods that incorporate full multidisciplinary coupling/optimization. The approximate schedule for the 3 steps is to accomplish the goals in 1, 2, and 5 years, respectively. Steps 1 and 2 are rather straightforward and involve developing interfaces and data management methods to utilize methods already in existence. The biggest obstacle is numerical modeling methods that permit rapid turnaround time. Step 3 is the tough one. Here is where the breadth versus depth dilemma is fully encountered or the "Brick Wall" as it was described by Sliwa and Abel in the 1988 Multidisciplinary Analysis and Optimization Symposium (reference 2). This dilemma is the subject of later discussion.

PROGRESS AND PROBLEMS

The first obstacle encountered by the HiSAIR team in attempting to perform as higher order analysis of a conceptual design was the time required to construct a refined numerical model from the coarse

surfaced grids traditionally used in conceptual design. This situation is no surprise to those familiar with the design process. The time required and the tools used in the first passthrough are shown in figure 3. Twelve weeks were required and a geometry enrichment system (ANVIL 5000) was used. Twelve weeks for numerical modeling is of course unacceptable for a design system. The Aeronautical Surface Typology Analysis Routines (ASTAR) box shown in the figure represents a collection of programs that allow a user to assemble a CFD surface grid. Functions such as spline fitting, surface intersections, and surface grid point spacing control are included in ASTAR. In response to this situation, as well as other user needs, Langley has created a geometry laboratory (GEOLAB). The GEOLAB and HiSAIR team have identified the methods and hardware to accomplish the situation shown in figure 4. The geometry enrichment will be accomplished in 1 day using an interactive version of ASTAR. The interactive ASTAR will also use elements ANVIL 5000 and SMART (Solid Modeling Aerospace Research Tool) to help achieve rapid turnaround time. SMART is a preliminary design surface geometry software package that has some very sophisticated user interfaces and was developed by the Space Systems Division at Langley. Both ASTAR and SMART software are in the public domain.

The success or lack of success of any multidisciplinary analysis, design, and optimization system, such as HiSAIR, depends on the data management system. In the HiSAIR team's approach to data management was an acute awareness, based on previous experience, that any system which placed an undue burden on the discipline researcher or required significant changes in his methods of data interface, was likely to meet with less than complete success. The data management of HiSAIR is being constructed around software developed by Langley referred to as EASIE (Environment for Application Software Integration and Execution) and commercial data base systems. EASIE provides an executive controller and a high-level data base manager. The EASIE executive function provides configuration management and an operating system free environment for data review/modification and program execution. These data management tools are intended to provide the framework for a design and analysis system that is flexible and in which the data management functions are as transparent as possible to the research engineer.

Understanding the data exchange is also critical to data management. HiSAIR initiated this process by surveying the data needs and availability for each discipline involved and compiling a catalog of available computer codes. The desire to incorporate higher order analysis codes more appropriate to today's more complex aircraft greatly increased the data needs and exchange. In addition to the form and format of the data, the most efficient flow of the data between each user is being examined. A data flow modeling tool referred to as DeMAID (Design Managers Aid for Intelligent Decomposition) is being used. DeMAID is a knowledge-based tool for multilevel decomposition of complex design problems and is reported in reference 3. The method not only determines the hierarchial structure of the information exchange, but helps prevent important interactions from being overlooked and performs a scheduling function that exploits parallel processing of modules or groups of modules to save time.

A result of a rather simple demonstration of DeMAID is shown in figures 5 and 6. DeMAID was used to study the data exchange between 19 computer programs used in the design of a supersonic airplane concept by the Vehicle Integration Branch at Langley. Although the data flow was well understood, prior knowledge was deliberately ignored to test DeMAID. The programs (or modules) were randomly assigned numbers and were ordered into an NxN matrix format as shown in figure 5. Data output is represented as a horizontal line and input is represented as a vertical line. The solid dots represent data that are output from one program and input to another, an interface. Any line leaving a program on the left or entering from below represents a feedback that implies an iteration. Programs with no input requirements are initialization programs whose inputs are satisfied from an outside source-- example mission requirements. The DeMAID tries to minimize the number of feedbacks. The results of the DeMAID reordering is shown on the right side of figure 6. The feedback loops were eliminated and the unidirectional data flow established. Note that eliminating all feedback loops is not typical and this will certainly not occur in a more sophisticated multidisciplinary design process, but DeMAID will

minimize the feedback loops by its grouping of the modules into circuits (see reference 3). The left side of figure 6 more clearly indicates the significance of the reordering of the programs. Program 14 and 7 initiate the process, then programs 11, 18, 10, 8, and 13 can be run simultaneously, and so on until program 16 completes the design. The 19 programs are grouped in 8 steps. Even for this simple example, the insight gained is valuable. Imagine, if you will, the larger much more complex multidisciplinary design problem faced by the HiSAIR team and the value of a tool such as DeMAID is obvious.

An interesting impediment to the HiSAIR goal of increasing multidisciplinary interaction was observed shortly into the project. Due to Langley's organization along discipline and even subdiscipline research lines, each organization had become independent in its general data needs. For instance, the various structures groups exercised aerodynamic codes to generate their own load distributions. There is nothing wrong with this, especially in the research mode, but it does fail to take advantage of the superior load definition and control surface effectiveness calculations afforded by advanced aerodynamic codes. If nothing else, HiSAIR has already succeeded in creating an increased awareness at Langley that a multidisciplinary understanding will help improve the methods employed within the disciplines.

Concurrent with establishing a multidisciplinary analysis process with unprecedented levels of discipline interaction and turnaround time, the HiSAIR team is researching the more difficult problem of multidisciplinary optimization. An integrated design exercise is being conducted utilizing a design process centered on nonhierarchial decomposition with optimization guided by system response derivatives. These methods were discussed by Sobieski in the 1988 Multidisciplinary Analysis and Optimization Symposium (reference 6) and more recently in reference 7. The integrated design exercise is currently focusing on HSCT wing design to develop mathematical methods for the integrated design of aircraft configurations. The integrated design exercise is more fully reported by in reference 4 which is a paper delivered elsewhere in this symposium.

PLANS

To achieve an advanced system design and analysis capability that can generate nonderivative aircraft concepts requires that the breadth versus depth dilemma be solved. Rigorous full system optimization methods require a well-defined continuous model. Engineering solutions to problems encountered are rarely continuous. For instance, the best solution to a stability and control problem may be to switch from a horizontal tail to a canard rather than to continue optimizing the horizontal tail. System optimization methods can find the best solution for a given set of technologies, but are not suited to establishing the technology set. The involvement of discipline experts in conceptual and advanced system design in real time is required to solve this dilemma. The exact form for a conceptual design system with global optimization capability has not been established, but several ideas have emerged.

A system such as that illustrated in figure 7 shows much promise. The figure illustrates a discipline expert in aerodynamics working to whatever level of detail that is appropriate to apply his full capabilities and insight. The expert is provided with multidisciplinary guidance and assistance by utilizing routines that provide information about the impact of the aerodynamic configuration on other disciplines. These multidisciplinary guidance and assistance routines could be knowledge based or expert systems incorporating artificial intelligence features as appropriate. They could also be simple extrapolation formulas based on the system response derivatives computed for the baseline. The guidance and assistance routines would be simpler than the methods normally employed by the experts in the respective disciplines. These simpler methods are most likely required to keep the system from bogging down due to either the level of detail required for full multidisciplinary coupling or lack of computer speed and storage. Experts from each of the involved disciplines can work real time in coordination with each other to evolve a set of configurations within a set of previously agreed guidelines. In addition to the configuration set, system sensitivity derivatives would be generated. The

output would be a configuration set and sensitivity derivatives that would form the input for the rigorous optimization methods previously discussed. This approach has the very attractive feature that allows equal discipline participation in the design process. The traditional sequence in which the discipline expert fixes problems generated by someone else no longer applies. Many details would have to be solved to make such a system practical but it appears feasible within the 5 year goal of HiSAIR.

BENEFITS

The HiSAIR project has the overall objective of being able to determine the best technical solution when integrating increasingly complex and intertwined technologies. Further, the philosophy is to develop a system that allows the discipline expert to use his physical insight and methods to solve system problems. This philosophy is reflected in the proposed methods of multidisciplinary coupling and in the optimization methods employed. The result will be not only better aircraft designs that achieve maximum advantage from new technologies, but the development of analytical methods and data bases that find more ready applications in the aeronautical community. Equally important is the developing multidisciplinary cultural awareness. This new culture for doing business in an integrated multidisciplinary design environment is already stimulating new ideas and research. As the HiSAIR project proceeds past the first stage to the tougher problems of multidisciplinary optimization, the interactions and the new research stimulated will increase.

CONCLUDING REMARKS

NASA Langley Research Center has embarked on an organized program (HiSAIR) to strengthen multidisciplinary aircraft analysis, design, and optimization. The project cuts across the organizational discipline lines of the Center and uses the High-Speed Civil Transport as a research focus. The project has met with an enthusiastic response from most of the discipline organizations. The discipline specialist perceives HiSAIR as a means to broaden his or her knowledge of and influence on the aircraft design process, as well as stimulate new research with increased relevance to the aeronautical community beyond their specialist area. The long-term strategy for HiSAIR is to proceed through three steps. Step 1 is to establish a high-fidelity rapid multidisciplinary analysis process; step 2 is to develop a local optimization design process using iterative and system sensitivity derivatives methods; and step 3 is to develop a global optimization system with full multidisciplinary coupling/optimization.

HiSAIR is approximately half way to the accomplishment of step 1. The major obstacles encountered in establishing a high-fidelity rapid multidisciplinary analysis system were not unexpected. Geometry or numerical modeling and data management were the two principal concerns. A number of existing computer programs as discussed in the text are being assembled and modified for the modern workstation environment to handle these concerns. Data flow modeling tools that use artificial intelligence techniques to decompose complex systems are being employed to understand the nature and most efficient processing of the data. These decomposition tools along with system sensitivity analysis methods are being researched as tools for the optimization steps of HiSAIR. Methods of coping with the breadth versus width are evolving and are under study and must be developed to accomplish the global multidisciplinary coupling/optimization goal. Equally important is that an atmosphere is developing in which the discipline specialist has an awareness and better appreciation for impact of their decisions on the overall aircraft system. The major benefit to the Langley Research Center as a research organization is the stimulation of new research ideas within the disciplines that lead to methods and technology more applicable to our rapidly increasingly complex world of aircraft design.

REFERENCES

1. Aeronautics Technology Possibilities for 2000: Report of a Workshop. Workshop on Aeronautical Technology: A Projection to the Year 2000, Aeronautics and Space Engineering Board, Commission on Engineering and Technical Systems, National Research Council, National Academy Press, Washington, DC, 1984.

2. Sliwa, Steven M.; and Abel, Irving: Overview of Dynamics Integration Research (DIR) Program at Langley Research Center--Goals and Progress. NASA Conference Publication 3031, Part 1, Advances in Multidisciplinary Analysis and Optimization, Proceedings of a symposium held in Hampton, Virginia, September 28-30, 1988.

3. Rogers, James L.: A Knowledge-Based Tool for Multilevel Decomposition of a Complex Design Problem. NASA TP-2903, May 1989.

4. Barthelemy, J.-F. M.; Coen, P. G.; Wrenn, G. A.; and Riley, M. F.: Application of Multidisciplinary Optimization Methods to the Design of a Supersonic Transport. Proposed paper in the Third Air Force/NASA Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, San Francisco, California, September 24-26, 1990.

5. Wrenn, G. A.; and Coen, P. G.: Development of an Efficient Aeroelastic Analysis and Sensitivity Analysis Capability for a Supersonic Transport Aircraft. Proposed paper in the Third Air Force/ NASA Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, San Francisco, California, September 24-26, 1990.

6. Sobieszczanski-Sobieski, Jaroslaw: Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems. NASA Conference Publication 3031, Part 1, Advances in Multidisciplinary Analysis and Optimization, Proceedings of a symposium held in Hampton, Virginia, September 28-30, 1988.

7. Sobieszczanski-Sobieski, Jaroslaw: Multidisciplinary Optimization for Engineering Systems: Achievements and Potential. NASA TM-101566, March 1989.
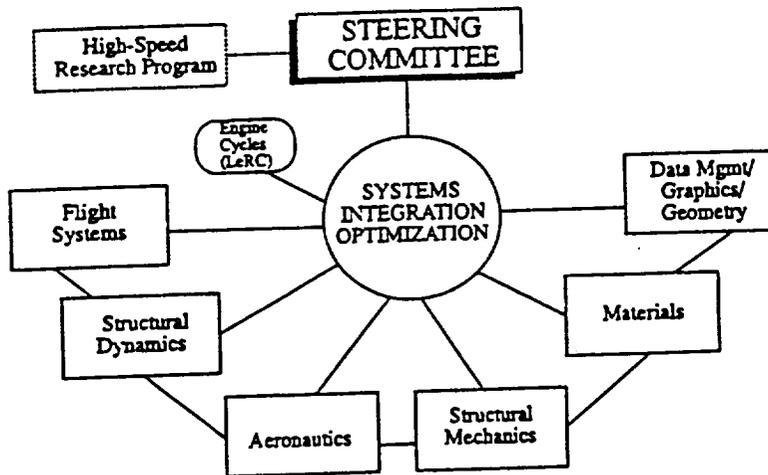
Figure 1. - High-Speed Airframe Integration Research Management
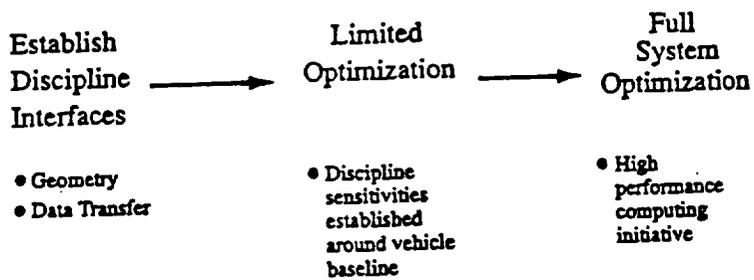
Figure 2. - Long-Term Multidisciplinary Airframe Integration Strategy
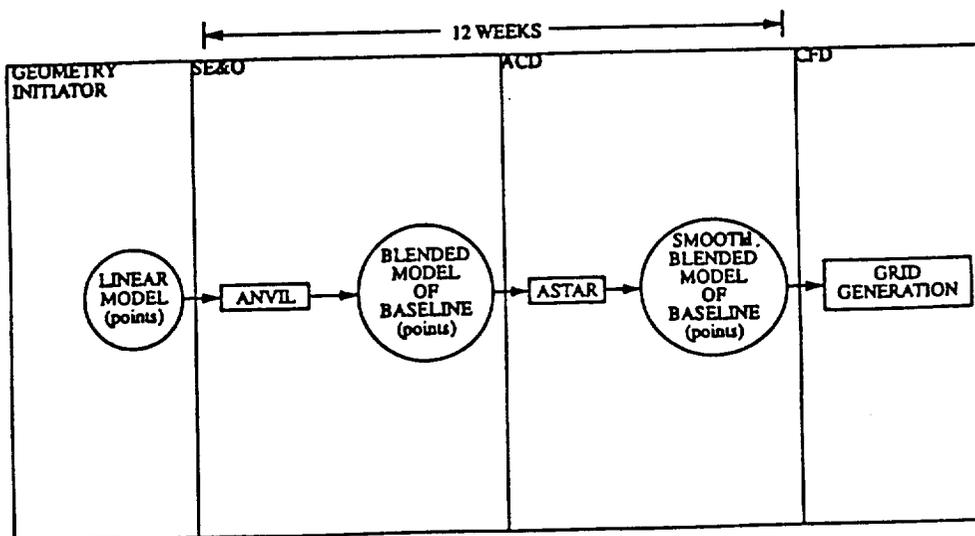
Figure 3. - Current Geometry Process
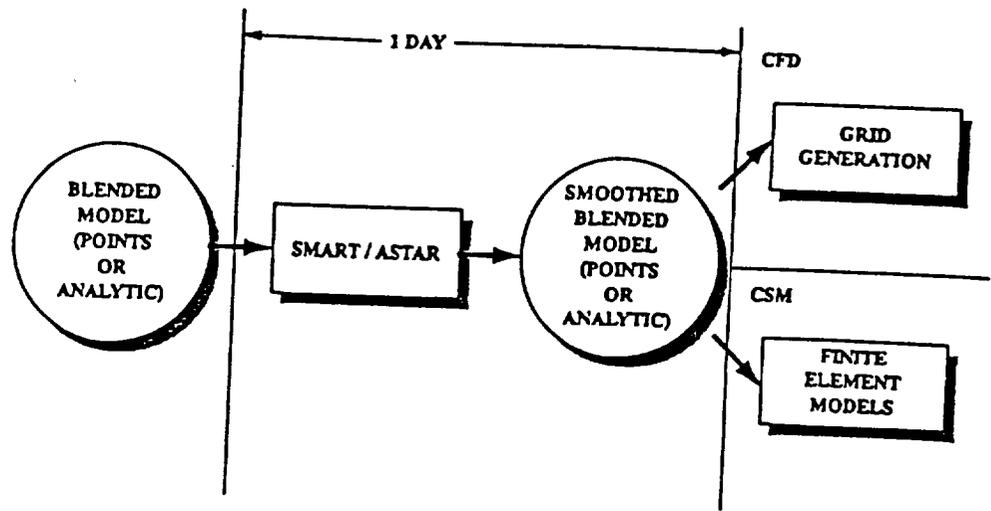
Figure 4. - Advanced Geometry Process
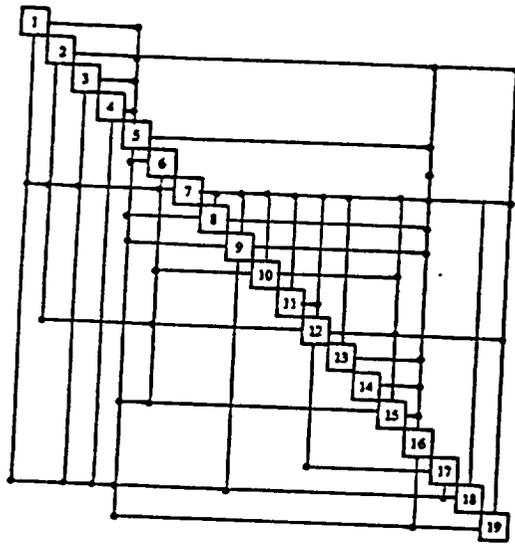
Figure 5. - Initial N² - Matrix for VIB Design Process
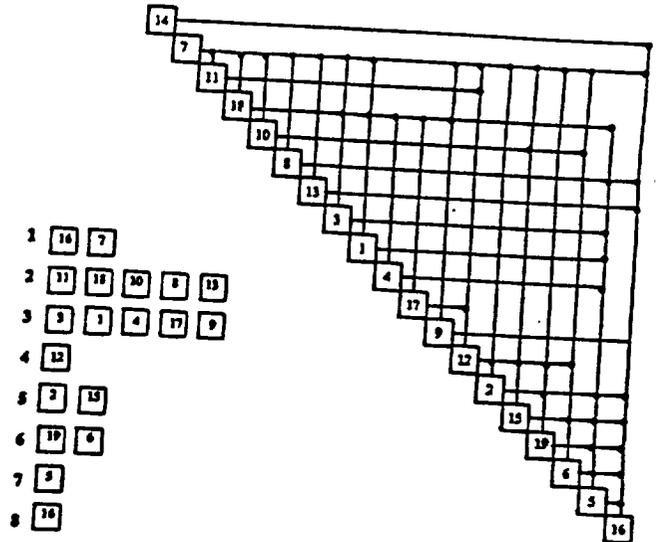
Figure 6. - N² - Matrix for VIB Design Process After Organization
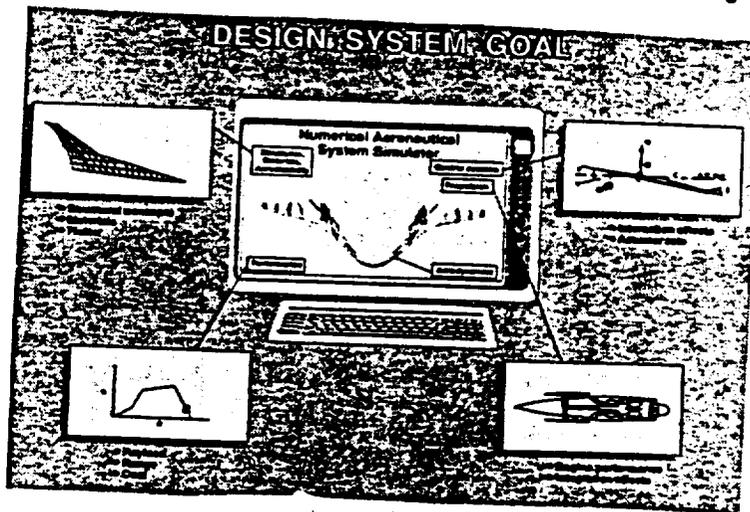
Figure 7. - Multidisciplinary Design System

411

C-6.

MULTIDISCIPLINARY HYPERSONIC CONFIGURATION OPTIMIZATION

M. Levine, H. Ide, and S. Hollowell
Rockwell International
North American Aircraft
Los Angeles, CA 90009

N94- 71473

ABSTRACT

Hypersonic vehicle design involves several complex, highly coupled disciplines. The need to use multidisciplinary optimization techniques to determine the optimal configuration is rather apparent. This paper presents a multidisciplinary configuration optimization technique which directly applies to the very difficult challenge of hypersonic vehicle design.

INTRODUCTION

The developments in computational fluid dynamics (CFD) techniques in recent years are significant. Although CFD analysis can require significant time to be performed due to flow complexity and grid issues, the accuracy with respect to high speed vehicle performance is very reliable. Concurrently, various types of optimization methods based on CFD and CSM (computational structural mechanics) analyses are recieving more attention, references 1-4.

In the case of hypersonic vehicles, the use of multidisciplinary optimization techniques is very important for proper sizing of the vehicle. The key disciplines are aerodynamics, inlet performance, propulsion, and structures. One of the serious concerns in the preliminary design stage is the vehicle performance based on an assumed takeoff gross weight. Particulary, effective specific impulse is an important parameter in the attainment of a desired vehicle trajectory. Since the specific impulse is a direct function of the vehicle thrust and drags, minimizing the drag forces is essential. On the other hand, the same thing can be achieved by maximizing thrust through the improvement of the inlet perfomance and the fuel volumetric efficiency.

The traditional sizing method attains closure on a vehicle design by photographically scaling the baseline to achieve the required fuel fraction. This occurs at the intersection of the fuel required and the fuel available curves,as shown in figure 1. It is possible to attain closure by bringing the fuel required curve down and/or by bringing the fuel available curve up. As a first step, the fuel available will be held constant in the approach in this paper. Hence, the approach in this paper will concentrate on bringing the fuel required curve down. For simplicity, this will be accomplished by maximizing the specific impulse (Isp) at a critical point in the trajectory by modifying the baseline configuration geometry.

In this paper, a specific approach to sensitivity calculation, the determination of independent design parameters, the formation of a global sensitivity matrix, the optimization process, and some results will be shown.

PROBLEM STATEMENT

In order to excercise CFD and other derived sensitivities in an optimization problem, a vehicle specification and a key point in the trajectory are defined. The hypersonic vehicle baseline configuration used in this study is shown in figure 2, including the design variables chosen. Presently, we are dealing only with the forebody of the vehicle for simplicity. Therefore, the effect of the forebody on the aftbody aerodynamics is neglected at this time. The aftbody (ie., nozzle, bodyflap, elevon, etc.) effects are included in the propulsion contributing analysis (CA) and the trim constraint.

The flight condition at which the vehicle will be optimized is as follows; Mach = 16, q = 1500 psf, and α = 0.0 degrees. Finally, the objective function for this optimization is defined as follows.

Objective: maximize I_{sp}

$$I_{sp} = \frac{T - D}{m_f} \tag{1}$$

$$D = \begin{cases} D_0 = D_{ram} + D_{fb} + D_{visc} + D_{aft} \; ; \text{UNTRIMMED} & \text{(2a)} \\ D_0 + D_e + D_{bf} \; ; \text{TRIMMED} & \text{(2b)} \end{cases}$$

where,

T = thrust
D = drag
m_f = fuel flow rate
D_0 = total untrimmed drag
D_{ram} = ram drag
D_{fb} = forebody pressure drag
D_{visc} = forebody viscous drag
D_{aft} = aftbody drag (held constant)
D_e = elevon drag
D_{bf} = bodyflap drag

412
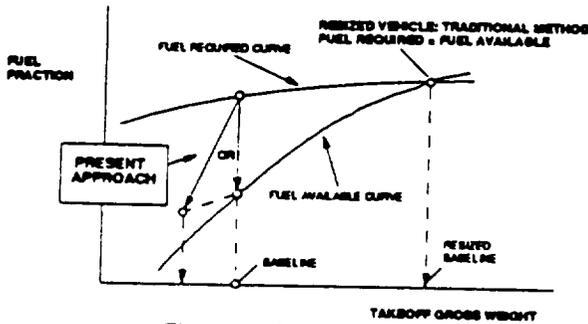
Figure 1. The sizing process
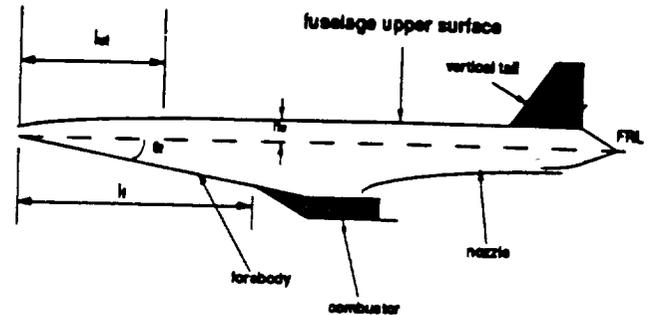
Figure 2. Baseline configuration

OVERALL OPTIMIZATION PROCEDURE

A multidisciplinary decomposition/optimization technique is used to develope the global sensitivities needed by the optimizer in an efficient manner. This technique involves four steps as shown in figure 3. (1)Determine the most efficient decomposition of the design process, by identifying the individual contributing analyses (CA's) that makeup the design process using an N-squared diagram. (2) Define the linkages between CA's by adding to the N-squared diagram developed in step 1 . (3) Calculate the sensitivity derivatives for each CA independently. (4) Combine the sensitivities in the global sensitivity equations (GSE) to determine the global sensitivity derivatives (GSD) which are then used by the optimizer to determine an optimum configuration. Further background and examples of applications of the methodology can be found in references 5-6. The following sections describe each step in more detail.

1. DECOMPOSE DESIGN PROCESS

The N-squared diagram, shown in figure 4, is a handy tool for presenting the functional decomposition and the linkages between CA's, which are discussed in this and the next section, respectively. It shows some of the disciplines included in the design process.

Each box is a contributing analysis card which contains information about CA. It identifies figures of merit, constraints, and control variables. It also defines the programs to be used to generate the sensitivity data and the person(s) responsible for running them.For the present work, the maximum amount of disciplines used in the results are those shown in figure 5. Results are also presented for optimizations where some of these CA's are not included.

2. DEFINE LINKAGES

Each oval in figure 4 identifies a connection between disciplines. By following the lines away from the oval and towards the CA boxes, the two disciplines involved in the linkage can be identified. Each oval is a data card which identifies the information passed from one CA to another CA. It also defines the person(s) responsible for generating the data and the person(s) that would recieve the data during a traditional design cycle. The linkages that occur in the present work appear as dots in figure 5.

3. CALCULATE LOCAL SENSITIVITY DATA

Each discipline independently calculates the sensitivity derivatives which are defined during step 2. The sensitivity

Figure 3. Overall optimization procedure

413
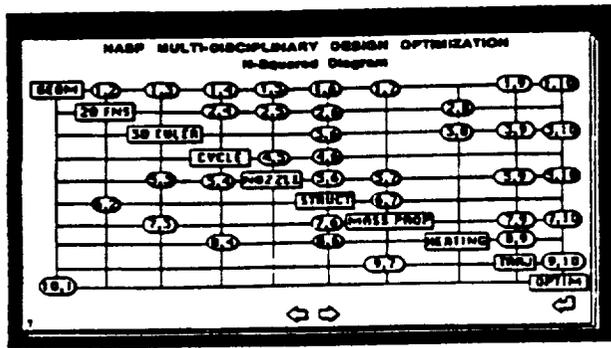
Figure 4. N-squared diagram example

data is calculated using a finite difference approach. Three different ways of determining the sensitivity data and the change in the sensitivity data away from the baseline are presented in figure 6.

The first-order one-sided difference (FO-OSD) approach has the key advantage of requiring only the baseline and one additional run to calculate the first- order sensitivity data. However, the accuracy in the direction opposite to the perturbation can be very poor if the curve is not close to be linear.

The first-order central difference (FO-CD) requires the baseline plus two runs to calculate the sensitivity data. It produces better accuracy than FO-OSD in one direction, but gives up some accuracy in the other direction which makes this approach not worth the extra cost of the additional run.

However, without making any additional runs, than the FO-CD approach, the second-order (SO) approach can be used. The advantage to the SO approach is the ability to model the nonlinearity in the sensitivity data. The FO-OSD approach and the SO approach each have their advantages and disadvantages, which one is best to use depends on the nonlinearity of the problem.

Figure 5. Step 2 for present work

The use of pre/postprocessors in order to speed up the preparation of input data for the aerodynamic and structural flexibilty analysis was very important in generating the local sensitivity data in a timely fashion. The diagram in figure 7 illustrates and describes those used in the present work.

4. SOLVE GLOBAL SENSITIVITY EQUATIONS AND PERORM OPTIMIZATION

The N-squared diagram, in figure 5, translates into the set of global sensitivity equations presented in figure 8. Any of the disciplines shown can be neglected by removing the proper rows and columns from the matrix equation. For example, in order to remove the effect of the 2-D Navier-Stokes CA from the optimization, the first row and first column would be deleted.

The right-hand side (RHS) of the equation deals with the local sensitivity of the outputs from each CA with respect to a single design variable. If the design variable is not a direct input to a particular CA, then all the local sensitivities in the RHS are zero with respect to that variable for that CA as shown in figure 8 for the propulsion CA.

The left-hand side (LHS) includes the cross-coupling (or linkage) sensitivity matrix and the solution vector. The

Figure 6. Sensitivity calculation
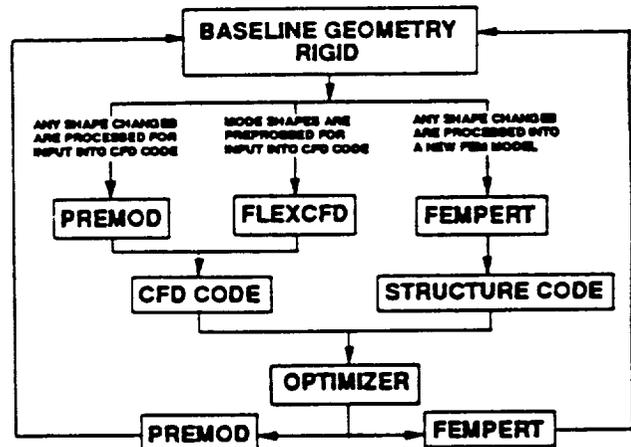
Figure 7. Pre/postprocessors

414
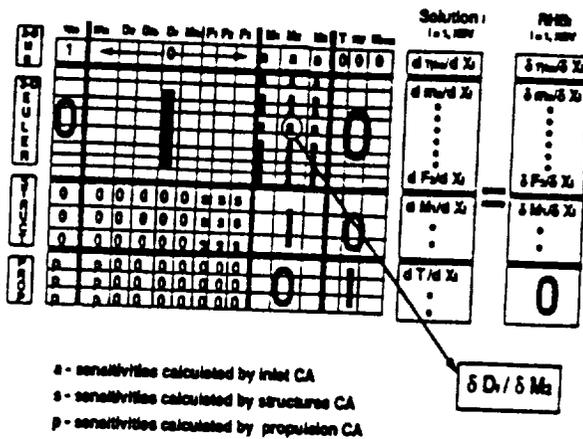
Figure 8. Global sensitivity equations

solution vector contains the global sensitivity of the outputs from each CA with respect to a single design variable. There will be one RHS and one solution vector for each design variable. It is important to note that although the local sensitivities on the RHS are zero for a particular CA with respect to a particular design variable the global sensitivities need not be zero.

The resulting global sensitivity data is then used to update the vehicle performance during the optimization run. This is accomplished using a Taylor series expansion for each of the outputs from each of the CA's. The objective function and the constraints are then updated and passed to the optimizer.

The optimizer used in the present work is the ADS (Automated Design Synthesis) program, reference 7, with ISTRAT=0, IOPT=4, and IONED=7. ISTRAT=0 indicates that no initial strategey is used. The use of the Method of Feasible Directions (MFD) for constrained minimization, references 8-9, is indicated by IOPT=4. The one-dimensional search, IONED=7, finds the minimum of an constrained function by first finding bounds and then using polynomial interpolation.
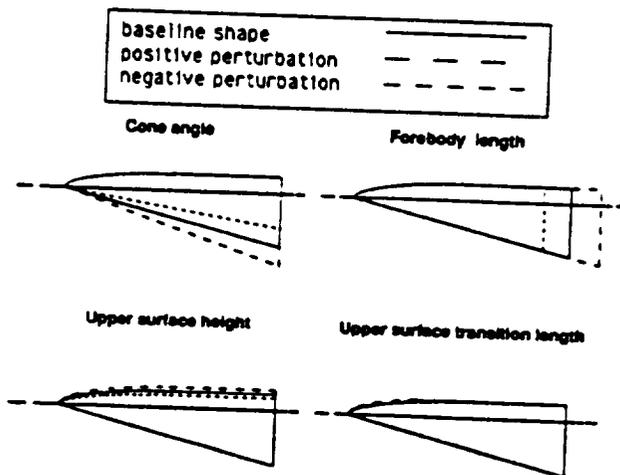
Figure 9. Design variables

CONFIGURATION DECOMPOSITION

The configuration is decomposed into independent design variables which are used by the optimizer to improve the vehicle performance. The design variables which will be used in the hypersonic forebody optimization example in this paper are shown in figure 9. The main concern when decomposing a configuration is to make the design parameters ass independent as possible. The more independent they are the larger the allowable move limit in the optimization run, which can possibly reduce the number of optimization cycles. At the beginning of each cycle, the final shape from the previous cycle is analyzed and new global sensitivities are generated. By reducing the number of optimization cycles, a substantial amount of computational cost and time can be saved.

RESULTS AND DISCUSSION

In this paper, four optimization cases are examined. The objective function for the first two cases uses equation 2a in conjunction with equation 1. These equations produce an untrimmed specific impulse (Isp). Cases 3 and 4 use equation 2b instead of 2a to produce a trimmed Isp for the objective function. All four cases have a forebody tank volume constraint for simplicity and to concentrate on one aspect of the present approach's capability. The design variables are limited to a maximum of 10% change, plus or minus, from the baseline values. Each case adds either another CA or an additional constraint to the optimization process. More details and results of each case are discussed in the following paragraphs.

CASE 1

The initial optimization case shown in this paper contains the CA's, 3-D Euler and propulsion. As a first step, the objective is to maximize an untrimmed Isp, neglecting viscous effects on inlet performance, trim effects, and flexibility effects. These additional effects will be added one at a time into the cases to follow. The only constraint for this case is the forebody tank volume constraint, which is defined as follows:

$$-\varepsilon < (\text{present tank volume - baseline tank volume}) < \varepsilon \quad (3)$$

$$\varepsilon = 0.1 \text{ \% of the baseline tank volume}$$

The results of case 1 are shown in figure 10. The most significant design variable change occurs to the geometric transition length (DV-4). The increase in this design variable decreases the tank volume slightly, but it significantly decreases the forebody drag which is a key factor in maximizing the Isp. The loss in volume is compensated by the other varaibles. The forebody length (DV-1) increases while the cone angle (DV-2) decreases to produce a more slender forebody which helps maintain the tank volume while reducing the drag. The upper surface height (DV-3) paramater is relatively ineffective, although it increases slighty to help maintain the tank volume. The optimizer predicts approximately a 17% increase in the untrimmed effective Isp.

415

Optimization Parameter	Optimization Results*
Design Variable	
1 - Forebody length	1.0562
2 - Cone angle	0.9087
3 - Upper surface height	1.0126
4 - Geometric transition length	1.1000
Objective	
Effective Isp	1.1693

* Normalized by baseline values, baseline effective Isp is untrimmed
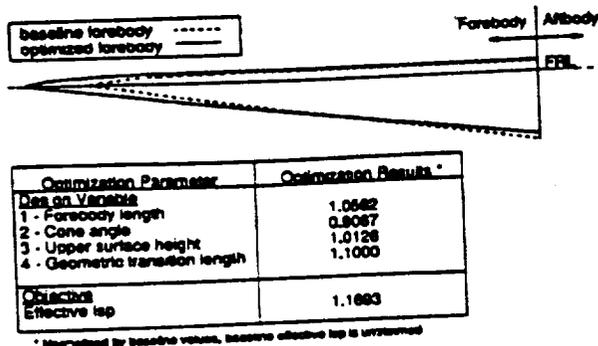
Figure 10. Case 1 results - initial optimization

Optimization Parameter	Optimization Results*
Design Variable	
1 - Forebody length	1.0375
2 - Cone angle	0.9367
3 - Upper surface height	1.0153
4 - Geometric transition length	1.1000
Objective	
Effective Isp	1.1681

* Normalized by baseline values, baseline effective Isp is untrimmed

Figure 11. Case 2 results - addition of 2-D Navier-Stokes contributing analysis to case 1

CASE 2

The second optimization case is the same as case 1 with the addition of the 2-D Navier-Stokes CA. This CA contributes the viscous effects on inlet performance to the optimization process.

The results for case 2 show that the changes in the design variables are qualitatively similar to case 1, as shown in figures 10-11. DV-3 and DV-4 have almost identical changes in magnitude in cases 1 and 2, which is due to the fact that these two variables have an insignificant effect on inlet performance. However, the magnitude of the changes in DV-1 and DV-2 are smaller for case 2. This indicates that the benefits of making the forebody more slender reaches a maximum closer to the baseline shape when the viscous effects on the inlet performance are included in the process. Even with these differences in magnitude, the objective function value is almost identical in the two cases. This is due to a positive viscous effect on the inlet performace due to the change in DV-1 and -2, which counter-balances the increase in drag for the less slender forebody.

CASE 3

The third optimization case uses equations 1 and 2b to produce a trimmed Isp for the objective function. It includes the same CA's as case 2 with an added constraint. The constraint added is for maintaining trim given a trimmed baseline vehicle, and it is defined as follows:

$$-\varepsilon < \text{(summation of changes in moments)} < \varepsilon \quad (4)$$

$\varepsilon = 1\%$ of the baseline summation of moments

In order to trim the vehicle, the elevon deflection (DV-5) and the bodyflap deflection (DV-6) must be included as design variables. For this paper, these control surfaces produce only a moment and a drag. For simplicity, the effect of the bodyflap on the propulsion is neglected. The moments are used to satisfy equation 4, and the drags, which appear in equation 2b, are the performance penalty for trimming the vehicle. In order to start the optimization, a baseline condition for the control surfaces is required. The baseline condition for this paper is a 5.0 degrees deflection for both surfaces. These deflections add drag to the baseline, which creates a lower baseline Isp than the untrimmed cases.

The changes in DV-1 through -4 are similar to case 2, as shown in figure 12. It is important to note that the 20 % increase in the objective function is with respect to a trimmed baseline Isp. The percent changes in the previous cases are with respect to an untrimmed baseline Isp, which is larger than the trimmed value due to not including the drag from the control surfaces. Therefore, the actual value of the objective may be larger for case 2 than it is for case

Optimization Parameter	Optimization Results*
Design Variable	
1 - Forebody length	1.0339
2 - Cone angle	0.9480
3 - Upper surface height	1.0095
4 - Geometric transition length	1.1000
5 - Elevon deflection	0.8820
6 - Bodyflap deflection	1.0020
Objective	
Effective trimmed Isp	1.2053

* Normalized by baseline values, baseline effective Isp is trimmed

Figure 12. Case 3 results - addition of trim constraint to case 2

Optimization Parameter	Optimization Results*
Design Variable	
1 - Forebody length	1.0209
2 - Cone angle	0.9693
3 - Upper surface height	1.0029
4 - Geometric transition length	1.0760
5 - Elevon deflection	0.8820
6 - Bodyflap deflection	1.0320
Objective	
Effective trimmed Isp	1.1258

* Normalized by baseline values, baseline effective Isp is trimmed

Figure 13. Case 4 results - addition of flexibility to case 3

3. Of the two additional variables only the bodyflap seems to be effective. The optimized shape actually requires less bodyflap deflection than the baseline which reduces the drag due to this control surface. This accounts for most of the increase in the percentage change in the objective.

CASE 4

The fourth optimization case is the same as case 3 with the addition of the structures CA. This CA contributes the forebody flexibility effect to the optimization process. For simplicity, the structural flexibility CA is only linked to the 3-D Euler CA, and it is not presently linked to the 2-D Navier-Stokes CA,as seen in figure 5. Hence, the effect of flexibility on the viscous part of the inlet flow is neglected. The procedure for adding structural flexibility to the optimization process is described in more detail in reference 6.

In the previous cases, the optimizer produced a longer and more slender forebody. By adding the flexibility effect into the optimization process, the same type of shape occured, except that the design variables remained closer to the baseline values, as seen in figure 13. This indicates that the benefits of making the forebody longer and more slender reaches a maximum closer to the baseline shape when the flexibility effects on the forebody aerodynamics and the inviscid inlet performance are included in the process. For a rigid vehicle when the optimizer increases the forebody length and decreases the cone angle, the forebody drag is reduced. However, for a flexible vehicle these changes also generate larger deflections due to the air loads, which produces more drag compared to treating the vehicle as rigid. Eventually, the optimizer reaches a point where an additional increase in DV-1 and a decrease in DV-2 creates more additional drag due to flexibility than the decrease in drag due to the design variable changes. This explains why the change in the design variables and the improvement in the trimmed specific impulse is much smaller in case 4 than it is in case 3.

CONCLUDING REMARKS

A multidisciplinary configuration optimization technique which directly applies to the very difficult challenge of hypersonic vehicle design is presented and demonstrated. A simple hypersonic forebody design problem is used as an application of the technique. The basic result of the four optimizations is that a longer and more slender forebody produced a higher specific impulse. It is interesting to note that qualitatively the changes in the forebody design parameters are similar for all four cases.

By adding the 2-D Navier-Stokes CA to the initial optimization case, it is discovered that the inlet performance increased from viscous effects due to the design variable changes. However, this positive effect dropped off as the design varaibles got farther away from the baseline, which is evident from the case 2 results.

The most interesting result of adding the trim constraint is that the forebody shape changes (DV-1 through -

4) are almost identical to the previous case. In addition, the shape changes actually required trim moment from the control surfaces, which also reduced the trim drag. For the problem presented, only the bodyflap was effective in maintaining the trim constraint.

The result of adding the flexibilty is the same as result in reference 6. The addition of the other disciplines had no qualitative effect on the structural effect on the optimization results.

Future work will investigate optimizing for multiple design points. This can be accomplished by adding the trajectory contributing analysis. In addition, the optimization of the rest of the vehicle's shape (ie., aftbody, wing, inlet, etc.) needs to be included in the process.

REFERENCES

1. Abdi, F. F., Ide, H., and Shankar, V. J., "Optimization of Nonlinear Aeroelastic Tailloring Criteria," ICAS-88-5.7.3, 16th ICAS Conference, Jerusalem, Israel, 1988.

2. Ide, H., Abdi, F. F., and Shankar, V. J., "CFD Sensitiivity Study for Aerodynamic/Control Optimization Problems," AIAA Paper No. 89-2336, 29th SDM Conference, Williamsburg, VA, April 1988.

3. Ide, H. and Levine, M. S., "Use of Second-Order CFD Generated Global Sensitivity Derivatives for Coupled Problems," AIAA Paper No. 89-1178, 30th SDM Conference, Mobille, AL, April 3-5, 1989.

4. Giles, G. L., Tatum, K. E., and Foss, W. E., "Effects of Nonlinear Aerodynamics and Static Aeroelasticity on Mission Performace Calculations for a Fighter Aircraft," presented at Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, Hampton, VA, September 1988.

5. Sobieski, J. S., "On the Sensitivity of Complex, Internally Coupled Systems," AIAA Journal, January 1990.

6. Abdi, F. F., Ide, H., Levine, M. S., and Austel, L., "The Art of Aerospace Design: A Multidisciplinary Challenge," presented at Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, Hampton, VA, September 1988.

7. Vanderplaats, G. N., Sugimoto, H., and Sprague, C. M., "ADS-1: A New General-Purpose Optimization Program," AIAA Journal, Vol. 22, No. 10, October 1984.

8. Zoutendijk, M., Methods of Feasible Directions, Elsevier Publishing Co., Amsterdam, 1960.

9. Vanderplaats, G. N., and Moses, F., "Structural Optimization by Methods of Feasible Directions," Journal of Computers and Structures, Vol. 3, Pergamon Press, July 1973, pp. 739-755.

NUMERICAL DERIVATIVE TECHNIQUES FOR TRAJECTORY OPTIMIZATION

Wayne P Hallman

The Aerospace Corporation

El Segundo, CA

Introduction

The adoption of robust numerical optimization techniques in trajectory simulation programs has resulted in powerful design and analysis tools. These trajectory simulation/optimization programs are widely used, and a representative list includes the GTS system (Ref. [1]), the POST program (Ref. [2]), and newer collocation methods such as OTIS (Ref. [3]) and FONPAC (Ref. [4]) . All of these programs rely on optimization algorithms which require objective function and constraint gradient data during the iteration process. However, most trajectory optimization problems lack simple analytical expressions for these derivatives. In the general case a function evaluation involves integrating aerodynamic, propulsive and gravity forces over multiple trajectory phases with complex control models. With the newer collocation methods, the integration is replaced by defect constraints and cubic approximations for the state. While analytic gradient expressions can sometimes be derived for trajectory optimization problems, the derivation is cumbersome, time consuming and prone to mistakes. Fortunately, an alternate method exists for the gradient evaluation, namely finite difference approximations. In this paper some finite difference gradient techniques developed for use with the GTS system are presented. These techniques include methods for computing first and second partial derivatives of single and multiple sets of functions. A key feature of these methods is an error control mechanism which automatically adjusts the perturbation size to obtain accurate derivative values.

Numerical derivative methods for optimization applications have been studied elsewhere. In Ref. [5] a method for approximating Jacobian matrices based on balancing the truncation and rounding errors is presented. An error control method is proposed for forward difference gradient approximations of a single function in Ref. [6]. The methods developed in this paper share many ideas with these studies. A key difference is the technique developed for approximating the truncation errors during reperturbations.

The first derivative techniques considered below all use a central difference approximation requiring two additional function evaluations for each variable. While one sided differences can sometimes be used, it is difficult to choose an appropriate perturbation size without additional function information. In order to obtain this information, extra function evaluations are required which negates any computational savings of using a one sided partial. Also, in cases where several nonlinear functions are simultaneously differentiated using a common perturbation, the choice of the perturbation size is less critical with the more accurate two sided differences. The increased accuracy afforded with central differences is especially important during the final optimization algorithm iterations to obtain convergence.

Error Sources in Numerical Derivative Calculations

The special case for a single function f dependent on a single variable x is considered first. The function f along with its higher derivatives are assumed to be smooth. It is important to distinguish between the function f, and an approximation $\hat{f}$ which is computed by a numerical procedure. The mathematically defined function f is free of any error sources, and can be evaluated precisely. When a numerical scheme is used to approximate f, error sources which are inherent in the numerical computation arise. It is emphasized that the purpose of the numerical differentiation is to approximate the function f'. The function $\hat{f}'$, defined in the mathematical sense as a limit, is usually zero.

A constant ϵ_f, called the function accuracy tolerance, is assumed to exist such that

$$|\hat{f}(x) - f(x)| \leq \epsilon_f$$

for all x of interest. The value of ϵ_f depends on the computational technique and the number of digits, N,

retained in the computation. For simple analytic functions a good approximation is

$$\epsilon_f \approx |\hat{f}(x)|\epsilon_m$$

where ϵ_m is the machine precision ($\epsilon_m \approx 10^{(1-N)}$).

The central difference formula for the derivative of f is given by

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + T(x,h)$$

where $T(x,h)$ is the truncation error. This error term depends on the the third derivative of f and is equal to

$$T(x,h) = -\frac{h^2}{6}f^{(3)}(\xi)$$

for some point ξ in the interval $[x-h, x+h]$. When numerically evaluated values of f are substituted, the result is

$$f'(x) = \frac{\hat{f}(x+h) - \hat{f}(x-h)}{2h} + R(x,h) + T(x,h) \quad (1)$$

The term $R(x,h)$, called the roundoff error, combines the function accuracy errors at the two points and is bounded as

$$|R(x,h)| \le \epsilon_f/h \quad (2)$$

In practice the unknown error terms in Eq. (1) are dropped, and the derivative is approximated as the central difference using the $\hat{f}$ values.

Numerical Derivative Calculations With Error Control

To determine if the derivative approximation is suitable, a method for measuring and controlling the error terms is required. A performance index is defined as the sum of the errors, and the value of h is sought which satisfies

$$\min\{T(x,h) + R(x,h)\} \quad (3)$$

The error control mechanism is simply a means of adjusting the perturbation size h to approximately solve this problem. This technique is derived assuming an initial perturbation size h is given. The steps to approximate the errors and derive a new perturbation size h^* are outlined.

Error Estimates With Three Function Values

The bound from Eq. (2) is used to approximate $R(x,h)$. The truncation error term involving the third derivative of f can not be accurately approximated

with only three function values. Special assumptions must be made in this case to obtain estimates for this term. Following Ref. [5] if $f''(x) \ne 0$ and h is sufficiently small, it can be shown that

$$|f^{(3)}(\xi)| < \frac{6}{h^3}\left|f(x) - \frac{f(x+h) + f(x-h)}{2}\right| \quad (4)$$

Substituting $\hat{f}$ values, the truncation error is bounded as

$$|T(x,h)| < \left|\frac{\hat{f}(x+h) - 2\hat{f}(x) + \hat{f}(x-h)}{2h^3}\right|h^2 = kh^2 \quad (5)$$

Reperturbation Computations

Adding the two bounds, the sum of the truncation and roundoff errors is estimated as

$$\epsilon_B(h) = \epsilon_f/h + kh^2 \quad (6)$$

where k is obtained from Eq. (5). The value h^* which minimizes ϵ_B is given by

$$h^* = \left(\frac{\epsilon_f}{2k}\right)^{1/3} \approx \left(\frac{3\epsilon_f}{f^{(3)}}\right)^{1/3} \quad (7)$$

With the new value h^* computed, a test is applied to determine if reperturbations are actually required. In a neighborhood of the solution to (3) it is expected that the total error is insensitive to small changes in h. In addition since the error bounds are approximations, only a rough solution to (3) can be found in practice. The acceptance test requires a user supplied tolerance, $\rho > 1$, on the accuracy desired in the perturbation size. If

$$\rho^{-1} \le \frac{h^*}{h} \le \rho \quad (8)$$

then the central difference gradient approximation for $f'(x)$ with perturbation size h is accepted. Otherwise, h^* is the new value and additional function values are requested.

Refined Truncation Error Estimates With Four Function Values

The function is first evaluated at the point $x+h^*$. Using this new piece of information a better estimate for the truncation error can be obtained as follows. Define the quantities

$$\begin{aligned}
\Delta_1 &= f(x+h) - f(x) \\
\Delta_2 &= f(x-h) - f(x) \\
\Delta_3 &= f(x+h^*) - f(x) \\
a &= h^*/h
\end{aligned} \quad (9)$$

After manipulating Taylor series expansions, the third derivative of f can be shown to satisfy

$$h^3 f^{(3)}(x)/6 = \frac{-\Delta_1}{2(a-1)} + \frac{-\Delta_2}{2(a+1)} + \frac{\Delta_3}{a(a^2-1)} + Er \tag{10}$$

The error term Er depends on fourth derivatives of f times the term h^4. The third derivative estimate from Eq. (10) is used to approximate $T(x, h)$, and a new perturbation size h_t is obtained from Eq. (7). The new perturbation ratio, h_t/h, is then tested in Eq. (8). If this test is satisfied, then the perturbation size h is in fact acceptable, and reperturbations are not necessary. This check is performed to avoid cases where the truncation error estimates from Eq. (5) are erroneous because only three data points were used.

Refined Truncation Error Estimates With Five Function Values

When the ratio test fails, the function is evaluated at $x - h^*$ and an improved truncation error estimate is obtained using five function values. Following Eq. (9), define

$$\Delta_4 = f(x - h^*) - f(x)$$

From Taylor series expansions, it follows that

$$h^3 f^{(3)}(x)/6 = \sigma\left(-\Delta_1 + \Delta_2 + a^{-1}(\Delta_3 - \Delta_4)\right) + Er_3$$
$$h^4 f^{(4)}(x)/24 = \sigma\left(-\Delta_1 - \Delta_2 + a^{-2}(\Delta_3 + \Delta_4)\right) + Er_4$$
$$\sigma = 1/(2(a^2-1)) \tag{11}$$

The error terms Er_3 and Er_4 are proportional to h^5 times quantities involving fifth derivatives of f. Dropping the Er_3 term, a refined second order approximation to $f^{(3)}$ is found which in turn is used to estimate the truncation error $T(x, h^*)$. A new perturbation size h^{**} is then computed via Eq. (7) with the new error estimates. Once again the perturbation ratio test, Eq. (8), is performed, and if the test fails then the whole procedure is repeated with h^{**} as the new estimate. A maximum of five reperturbations are allowed.

The method outlined above always computes the derivative value with only two function values. It is possible during reperturbations to estimate the first derivative using the four function values available with a formula similar to Eq. (11). These higher order approximations are not utilized for the first derivative value since they have larger rounding errors and unknown truncation errors. In practice it has been observed that for a suitable value of h, the two point central difference formula usually yields acceptable derivative estimates for optimization applications.

Generalizations To Multiple Functions and Variables

The extension of the method to multiple variables is straightforward where the process is repeated for each independent variable. The case of multiple functions depending on a single variable is treated next. Assume that the n functions of a single variable x are given as

$$f_1(x), \ldots, f_n(x)$$

In most trajectory optimization problems the functions are evaluated simultaneously. Thus, for computational efficiency it is required to use a common perturbation size h for the n individual finite difference gradient approximations.

Ideally, function accuracy terms $\epsilon_f(1), \ldots, \epsilon_f(n)$ would be specified, and the rounding errors in each central difference derivative would be approximated as

$$R_i(x, h) \approx \epsilon_f(i)/h$$

A simpler method uses a common unit accuracy term ϵ_u for all functions where the errors are approximated as

$$R_i(x, h) \approx \epsilon_u |f_i|/h$$

The truncation error estimates $T_i(x, h)$ are approximated for each function as above.

The individual error terms are combined into a composite performance index utilizing weighting terms. Define the geometric average of the function derivatives, computed with central differences, as

$$\bar{f}' = \sqrt{\max\{|f_i'|\}} \sqrt{\max\{\min\{|f_i'|\}, \epsilon_m\}}$$

The weighting term for each function is given by

$$w_i = \left\{ \begin{array}{ll} 1 & \text{if } |f_i'| < \bar{f}' \\ 1/|\bar{f}'| & \text{if } |f_i'| \geq \bar{f}' \end{array} \right.$$

A performance index which measures the total weighted errors is

$$\bar{e}_B(h) = \sum_{i=1}^{n} w_i R_i(x, h) + \sum_{i=1}^{n} w_i T_i(x, h) = \ell/h + kh^2$$

where the ℓ and k terms are weighted sums of function accuracy errors and third derivative approximations similar to Eq. (6). The remaining computations follow the single function case. The parameter ρ is used to determine if reperturbations are necessary, and the truncation error estimates are refined using additional data when it is available.

Second Derivative Computations

The discussion so far was restricted to the computation of first derivatives. In postoptimality analysis applications, Ref. [7], it is necessary to compute numerical second derivatives of a function f dependent on an n-vector x. An error control technique for this problem is addressed next.

Once again the single variable case is treated first. It is assumed that a suitable perturbation size h_1 is available for the central difference first derivative of f. In postoptimality analysis this value is obtained from the just completed optimization problem. The central difference second derivative expression for $f(x)$ is

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + \frac{h^2}{12} f^{(4)}(\psi)$$

(12)

where $f^{(4)}$ is the fourth derivative of f evaluated at a point ψ in the interval $[x-h, x+h]$.

The roundoff error incurred when the numerically evaluated function values $\hat{f}$ are substituted is bounded as

$$R(x,h) \leq 4\epsilon_f / h^2$$

Following the same derivation for the central difference first derivative, the optimum perturbation size that minimizes the sum of the truncation and roundoff errors in Eq. (12) is

$$h_2 = 2 \left(\frac{3\epsilon_f}{|f^{(4)}|} \right)^{(.25)}$$

(13)

By combining Eqs. (7) and (13) and eliminating ϵ_f, it follows that

$$h_2 = 2 \left(\frac{f^{(4)}}{f^{(3)}} \right)^{(.25)} (h_1)^{(.75)}$$

(14)

To obtain an initial perturbation size for the second derivative calculation, the higher derivatives in Eq. (14) must be eliminated. If the third and fourth derivatives of f are assumed to be the same order of magnitude, then the optimal perturbation size for the second derivative calculation is

$$h_2 \approx 2(h_1)^{(.75)}$$

The second derivative computation begins by evaluating f at the points $x \pm h_2$ and forming the central difference approximation from Eq. (12). If no error control is desired, then the calculation is complete. Otherwise two additional function evaluations are performed at the points $x \pm 2h_2$. Using the five function

values now available, the fourth derivative of f can be estimated using Eq. (11). A new perturbation size, h_2^*, is then obtained which minimizes the truncation and roundoff errors from Eq. (13). The perturbation ratio h_2^*/h_2 is tested as in Eq. (8). If the ratio test fails, then h_2^* is the new perturbation size and two more function values are requested. The fourth derivative of f is then estimated using function values at the five points : $x, x \pm h_2, x \pm h_2^*$. This completes one cycle of the method. A maximum of five reperturbations are allowed.

The generalization to n variables first repeats the above sequence for each variable. From this data the diagonal elements of the Hessian with respect to x can be computed. The off diagonal Hessian matrix elements are obtained using a standard mixed derivative formula utilizing two additional function evaluations for each element. No error control is employed for the off diagonal elements due to the large computational expense that would be required. For most practical case if error control is used for the diagonal elements, the off diagonal Hessian elements computed with these same perturbation sizes are usually accurate.

Implementation Details

The numerical derivative methods outlined above have been implemented with the GTS system and also as stand alone FORTRAN. The interface is consistent with the reverse communication procedure described in Ref. [8]. Some implementation details are discussed below.

As noted in Ref. [5], the bound for $T(x,h)$ from Eq. (4) may be completely dominated by roundoff errors itself when h is too small. The truncation error estimate is totally unreliable in this case, and the perturbation size must be increased. The refined truncation error estimates using additional function values are also prone to catastrophic rounding errors if the perturbation sizes are too small, or if the ratio a is too small or large. In these cases, the perturbation size must be adjusted based on rounding errors only. In the second derivative calculations, the perturbation size is allowed to change by at most a factor of ten ($0.1 \leq a \leq 10.$) during a reperturbation. This results in more stable truncation error estimates.

Numerical first derivative computations are repeatedly performed during the optimization algorithm iteration. The finite difference perturbation sizes computed at one point are retained for the next derivative evaluation. In many problems it is sufficient to utilize the error control at the initial point only, and leave h fixed for all subsequent gradient evaluations. Options

are also available to compute the gradients with error control each time, or skip the error control procedure entirely. Recommended values for the reperturbation ratio factor ρ range from 3 to 10. Smaller values result in more reperturbations, and more accurate derivatives.

Finally, care must be taken in setting the function accuracy parameters $\epsilon_f(i)$. If they are chosen too large, then larger perturbation sizes result which can lead to inaccurate derivatives due to increased truncation error. If the function accuracy is underestimated, then rounding errors will dominate. One useful technique to approximate these parameters is to examine the optimization output from a similar problem if it is available. At the converged point, the active constraint errors give a rough estimate of the constraint function accuracies, (assuming tight convergence was obtained.) With feasible point methods, Refs. [8] and [12], intermediate points on the constraint surface are generated, and the function accuracy can be estimated as the smallest individual constraint errors observed.

Example : Optimal Control Space Shuttle Reentry

A shuttle reentry problem is used to illustrate the derivative techniques. This problem was originally analyzed in Ref. [10], and more recently investigated in Ref. [11]. The solution technique requires solving a two point boundary value problem utilizing the adjoint equations. An initial state vector at the reentry point with time $t_0 = 0$ is given as

Altitude	= 260000 ft
Longitude	= 0 deg
Latitude	= 0 deg
Velocity	= 25600 fps
Flight Path Angle	= -1.0 deg
Azimuth	= 90 deg

The desired conditions at the final free time t_f are specified as

Altitude	= 80000 ft
Velocity	= 2500 fps
Flight Path Angle	= -5.0 deg

Assuming a spherical earth model and simplified aerodynamics, it is desired to find the optimal control profile which maximizes the downrange (final longitude). The control profile is specified in terms of the angle of attack α with the bank angle β held equal to zero. The problem is formulated and solved using backward trajectory propagation where the vehicle is initialized at time t_f with the terminal constraints satisfied. The

eleven differential equations describing the state and adjoints are integrated backwards to time $t_0 = 0$, and the altitude, longitude, velocity, and flight path angle conditions specified above are imposed as constraints. An additional optimality condition on the final value of the Hamiltonian is imposed as the fifth constraint. The latitude and azimuth are held constant throughout the trajectory and eliminated from the problem.

The initial variable values and perturbation sizes chosen are listed in Table 1. The perturbation sizes were arbitrarily chosen as 10^{-3} to 10^{-5} times the initial variable values. The maximum downrange reentry problem was solved using fixed perturbation sizes and with two different derivative error control schemes. The first error control scheme used the method described in Ref [5] with the unit roundoff error estimated as 10^{-12}. The other case used the method described in this paper with a reperturbation ratio parameter $\rho = 5$. The function accuracy errors, $\epsilon_f(i)$ and constraint feasibility tolerances were chosen as

Constraint	$\epsilon_f(i)$	Tolerance
c_1 : Altitude at t_0	10^{-5}	10^{-4}
c_2 : Longitude at t_0	10^{-11}	10^{-8}
c_3 : Velocity at t_0	10^{-8}	10^{-5}
c_4 : Flight Path Angle at t_0	10^{-12}	10^{-7}
c_5 : Hamiltonian at $t_f = 0$	10^{-13}	10^{-5}

The optimization operator NLP3, Ref. [12], was used to solve the five equations in five unknowns. A standard Guass-Newton technique is employed. The results are summarized in Table 2 showing the total number of function evaluations (FE's), number of FE's for gradients, number of times reperturbations were performed, final root sum square of the constraint error, and the final perturbation sizes. The case utilizing the new error control method required reperturbations for the first four variables at the initial point. After this no additional reperturbations were performed. The case with the Ref [5] error control method required 73 function evaluations. The case without error control was terminated after five hundred function evaluations. None of the constraints were satisfied to within tolerance. The high degree of constraint nonlinearity and poor scaling can be seen by examining some derivative values. The constraint Jacobian matrix elements with respect to the first three variables ranged from 0.720 to $.416 \times 10^{11}$. The third partial derivatives of the constraints with respect to the first three variables ranged from -2.14×10^3 to -3.10×10^{24}.

The perturbation sequence for variable 1 and the resulting partials for constraint 2 are summarized in Table 3. At the initial point the truncation error for constraint 2 was dominant. The perturbation size was

Table 1 : Initial Variable Values and Perturbation Sizes

Variable	Initial Value	Perturbation Size
x_1 : Adjoint Altitude at t_f	−0.0110026242	10^{-6}
x_2 : Adjoint Velocity at t_f	−0.682174037	10^{-5}
x_3 : Adjoint Flight Path Angle at t_f	−0.906370867	10^{-5}
x_4 : Final Time (t_f)	3633.710745265 sec	10^{-1}
x_5 : Final Longitude	187.508990542 deg	10^{-1}

Table 2 : Comparison Results With and Without Error Control

	New Error Control	Ref [5] Error Control	No Error Control
Total Number of FE's	63	73	> 500
Number of FE's	5	5	> 47
Number of Perturbation FE's	58	68	> 460
Number of Reperturbations	4	9	0
Final RSS Constraint Error	$.47 \times 10^{-19}$	$.23 \times 10^{-16}$	$.42 \times 10^{-3}$
Final Pert Size for x_1	$.25 \times 10^{-9}$	$.10 \times 10^{-7}$	10^{-6}
Final Pert Size for x_2	$.90 \times 10^{-8}$	$.40 \times 10^{-7}$	10^{-5}
Final Pert Size for x_3	$.21 \times 10^{-7}$	$.79 \times 10^{-7}$	10^{-5}
Final Pert Size for x_4	$.51 \times 10^{-1}$	$.29 \times 10^{-6}$	10^{-1}
Final Pert Size for x_5	$.15 \times 10^{-1}$	$.014$	10^{-1}

Table 3 : Perturbation Sequence For Variable 1

Point	Δx_1	$\partial c_2/\partial x_1$	$T_2(x,h)$	$R_2(x,h)$	Total Rel Error ($\bar{\epsilon}_B(h)$)
1	$.100 \times 10^{-5}$	-102358.	45705.	0.00001	575558.
	$.213 \times 10^{-9}$	-97078.	0.00024	0.0470	0.0712
2	$.237 \times 10^{-9}$	-98063.	0.0028	0.0423	0.068
3	$.245 \times 10^{-9}$	-96755.	0.0032	0.0409	0.068
4	$.246 \times 10^{-9}$	-96750.	0.0029	0.0406	0.067

reduced once where the final derivative value was dominated by roundoff. Note that the initial truncation error estimate from Eq. (5) was 45704, while the actual change in the first derivative value, which also measures the truncation error in this case, was only 5280. The total relative weighted errors in the last column show stable behavior after the initial point.

The case with fixed perturbation sizes failed due to the inappropriate values chosen. This problem demonstrates the importance of carefully choosing these parameters for trajectory optimization applications if no error control is used. In practice the user should not be burdened with this choice. As the above results show the error control mechanism can automatically adjust these parameters and make a dramatic difference in the optimization algorithm performance.

To illustrate the second derivative techniques, a full second order sensitivity analysis was performed. A single perturbation parameter p was chosen as the ref-

erence surface area of the shuttle with nominal value $S_{ref} = 2690$ ft^2. The GTS postoptimality analysis operator, Ref. [7], was executed at the converged point. The Hessian of the Lagrangian function was computed with respect to the variables x and the parameter p. The function accuracy and reperturbation parameters were set equal to $\epsilon_f = 10^{-10}$, and $\rho = 5$. In this case all perturbations were performed with respect to the scaled variables, and the techniques described in Ref. [7] were used to obtain the final unscaled sensitivity data.

The Lagrangian Hessian diagonal elements required 11 sets of reperturbations. The fourth derivative estimates obtained from Eq. (11) ranged from $.97 \times 10^{-6}$ to $.20 \times 10^5$. The sensitivity derivative of the optimal final longitude, f^*, with respect to perturbations in S_{ref}, p,

was computed as

$$\frac{\partial f^*}{\partial p} = -0.1556 \times 10^{-2} \ \text{deg/ft}^2$$

This value was verified by solving perturbed optimization problems and found to be accurate to four significant figures.

Future Work

The finite difference methods considered in this report have been tailored for trajectory optimization applications where several nonlinear functions must be simultaneously differentiated using a common perturbation size. Experience has shown that to obtain consistently accurate derivatives and monitor the errors, two sided differences are required. There are instances, however, where one sided finite differences would suffice. At points far from the solution, optimization algorithms can make progress with one sided partials. If loose convergence tolerances are employed, then "convergence" can sometimes be obtained depending on the problem. However, if tightly converged answers are sought, then the gradient data must be computed more accurately as the solution is approached which requires central differences. A topic for future work is to extend the error control techniques to compute both one and two sided perturbation sizes. This could be done at the initial point, and the optimization algorithm could proceed with one sided partials until the errors become excessive. The more expensive two sided central differences could then be used to obtain final convergence.

Another topic of interest is to implement a method to estimate the function accuracy error terms $\epsilon_f(i)$ at the first point. Since the perfect mathematical problem model can not be evaluated in practice, this technique could only be approximate. However, based on the numerical model it should be possible to determine the noise level in $\hat{f}$. Finally, the use of variable scaling in conjunction with numerical differentiation is discussed in Ref. [6]. The second derivative techniques developed in Ref. [7] perform all gradient calculations with respect to scaled variables and functions. Utilizing scaling for first derivative computations is planned and should result in improved methods.

References

[1] "The Generalized Trajectory Simulation System," Volume 1-5, Report No. TR-0075(5549)-1, The Aerospace Corporation, El Segundo, CA, 30 June 1975.

[2] Brauer, G.L., Cornick, D.E., Olson, D.W., Petersen, F.M., and Stevenson, R., "Program To Optimize Simulated Trajectories (POST)," Volume 1-3, Report No. MCR-87-583, NAS1-18147, Martin Marietta Corporation, Denver, CO, September 1987.

[3] Paris, S.W., Hargraves, C.R., and Gailey, J.W. "Optimal Trajectories By Implicit Simulation," Volumes 1-4, Air Force Wright Patterson Aeronautical Laboratories Report, April 1988.

[4] Nguyen, H.N., "Optimal Solutions to Flight Mechanics Problems Using Nonlinear Programming And Collocation Techniques," Aerospace Corporation ATM No. 88(3501-06)-1, 12 August 1988. Not Available for External Distribution.

[5] Curtis, A.R. and Reid, J.K.,"The Choice of Step Length When Using Finite Difference Approximate Jacobian Matrices," J. Inst. Maths Applics (1974), Volume 13.

[6] Gill, P.E., Murray, W., and Wright, M.H., Practical Optimization, Academic Press, 1981.

[7] Hallman, W.P., "Sensitivity Analysis for Trajectory Optimization Problems," AIAA Paper 90-0471, AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV January 1990.

[8] Betts, J.T. and Hallman, W.P., "NLP2 Optimization Algorithm Documentation," Aerospace TOR-0089(4464-06)-1, August 1989.

[9] Betts, J.T. and Huffman, W.P., "A Parallel Trajectory Optimization Tool ," Paper 89-3613-CP, Proceedings of the AIAA Guidance, Navigation and Control Conference, August 1989.

[10] Zondervan, K.P., Bauer, T., Betts, J.T., and Huffman, W.P., "Solving the Optimal Control Problem Using a Nonlinear Programming Technique Part 3 : Optimal Shuttle Reentry," AIAA Paper 84-2039, AIAA/AAS Astrodynamics Conference, Seattle, WA, August 1984.

[11] Betts, J.T., "The Application of Sparse Broyden Updates in the Collocation Method For Optimal Control," Paper 88-4150-CP, Proceedings of the AIAA Guidance, Navigation and Control Conference, August 1988.

[12] Hallman, W.P., "NLP3 Optimization Algorithm Documentation," Aerospace IOC-A87-5752.5-45, September 1987. Not Available for External Distribution.

Integrated Aeroservoelastic Synthesis for Roll Control

Changho Nam[1] and Terrence A. Weisshaar[2]
School of Aeronautics & Astronautics
Purdue University
West Lafayette, Indiana 47907

Introduction

The objective of this study is to illustrate an integrated, parallel design procedure for optimal structural, aerodynamic, and aileron synthesis of an aircraft wing. The effects of combining weight minimization with structural tailoring (ply orientation and thickness) of a lifting surface, together with the wing geometry (sweep angle and taper ratio), and the aileron geometry (spanwise location and chordwise size) upon the lateral control effectiveness are discussed. Several optimization studies for the minimization of aileron hinge moment and wing weight, subject to a specified constant aircraft roll rate at a design airspeed(roll effectiveness) are performed.

Among the first historical aeroelastic stability and control problems encountered were those that dealt with the influence of wing flexibility on the roll response to asymmetrical aileron deflection. Wing twist due to the aileron deflection decreases the available aerodynamic rolling moment which, in turn, may cause aileron reversal (defined as the airspeed at which no rolling moment due to the aileron deflection is generated). Aileron reversal became a crucial design problem for World War II fighters for which high roll rates at high speeds were a prime combat maneuver. In general, the solution to aeroelastic difficulties was to increase the torsional stiffness or bending stiffness, a solution that led to weight increases that diminished performance in other areas[1].

The development of advanced composite materials provides a new dimension to aircraft design but also creates the need for extensive computer analysis. Advanced composites are advantageous not only because of their high strength-to-weight characteristics, but also because they allow the designer to alter and control aeroelastic deformation in a beneficial way[1]. Composites also have made previously impractical design options(such as forward swept wings) feasible. In addition, automatic control systems and their increased reliability provide a new way of achieving gust load alleviation, maneuver load control, ride control, and active flutter suppression.

Integrated, simultaneous design of an aircraft com-

posite material wing structure incorporating active control technology and structural optimization techniques requires a multidisciplinary perspective. To overcome the complexity of the design problem and expensive computation cost, procedures such as a multilevel decomposition technique and analytic sensitivity computation have been suggested by Sobieski[2]. Recently, the study of aeroservoelastic tailoring to exploit the interaction of aerodynamics, elastic structures, and controls was reported by Zieler and Weisshaar[3], Weisshaar and Nam[4], and Livne[5].

The present study is an extension of Reference [4]. Aileron hinge moment is the objective function to be minimized. The decrease in the aileron hinge moment can be converted into the wing weight reduction, because the size of the hydraulic actuator to deflect the aileron can be decreased. The ELAPS code[6, 7], a general equivalent plate analysis program developed by Giles is used for structural analysis. A constant load panel method is incorporated to calculate the aerodynamic forces of the wing and hinge moment required for aileron deflection. The wing planform is divided into 100 panel segments as shown Figure 1. In addition to determining the "best" configuration for roll requirements, the best spanwise location of the aileron for flutter suppression is found by using Nissim's aerodynamic energy concept[8] to determine whether the best spanwise aileron position for roll control is also acceptable for flutter suppression.

Aeroelastic Analysis

The roll effectiveness can be expressed in functional form(in terms of steady-state roll rate, p, and design airspeed, V) as follows[9]:

$$\left(\frac{pb}{2V}\right)/\delta_0 = \frac{M_R}{D_R} = \frac{Rolling\ power}{Damping-in-roll} \quad (1)$$

where b is wing span and the rolling power is:

$$M_R = (M_{R_{\delta_0}} + M_{F_{\delta_0}})/\delta_0 \quad (2)$$

$M_{R_{\delta_0}}$ and $M_{R_{\delta_0}}$ are the rigid rolling moment on the undeformed wing structure due to the aileron deflection δ_0 and the correction of the rolling moment due to wing flexibility, respectively. Further details about the roll effectiveness expression are contained in Reference [9].

[1]Graduate Student. School of Aeronautics & Astronautics.

[2]Professor. School of Aeronautics & Astronautics.

425

This roll effectiveness expression is an equality constraint for the design procedure. The requirement, $(pb/2V)_0 = constant$, provides the required aileron deflection angle $(\delta_0)_{req}$ from Equation 1 and determines the required aileron hinge moment.

The aileron hinge moment, H_{Total} is calculated using an aerodynamic panelling method and is the objective function for optimization. This hinge moment, H_{Total}, required for the roll maneuver is composed of two terms[9]:

$$H_{Total} = H_{Rigid} + H_{Flex} \qquad (3)$$

H_{Rigid} is the hinge moment required for the rigid(undeformed) wing structure and H_{Flex} is the correction to the hinge moment due to deformation. These terms are functions of the ply angle, skin thickness, and wing and aileron geometry. The expressions for the roll effectiveness, hinge moment were programmed and attached to the ELAPS code[9].
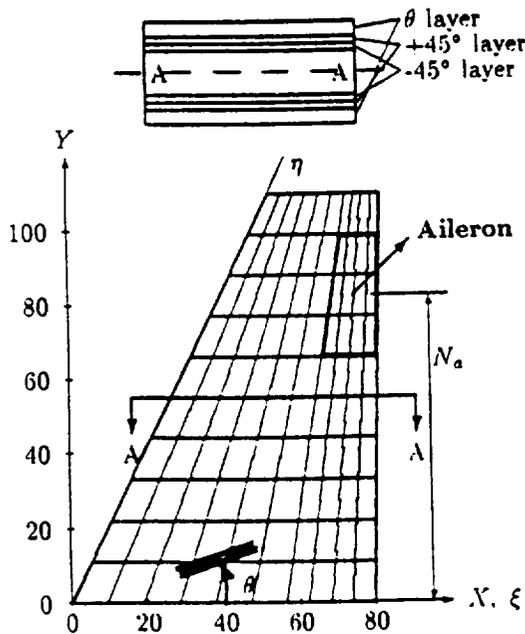
Figure 1: Wing planform geometry.

Optimization Examples and Results

The general optimization formulation is written as follows:

Minimize $H(x_i)$

subject to $l_j \leq g_j(x_i) \leq u_j, \quad j = 1, 2, \ldots m$

where x_i is a design variable vector and l_j, u_j are lower. and upper limits of constraints(such as requirements that wing tip twist does not exceed some

value). To solve the optimization problem, Rosenbrock's procedure[10] was used. For this procedure, the derivatives of the objective function with respect to design variables are not needed. However, this algorithm may be slow to converge.

Constraints are

1. $(pb/2V) = 0.08, \quad p \simeq 90°/sec.$

2. $|Wing\ Tip\ Twist\ | \leq 2.0°$

3. $15\% \leq N_a \leq 85\%$ of the semi-span

4. $0.2 \leq \bar{c}_{TE} \leq 0.4$

5. $-30° \leq \Lambda \leq 30°$

6. $0.25 \leq \lambda \leq 0.75$

The wing planform analyzed is shown in Figure 1. The design airspeed is 400 miles per hour at sealevel $(q=2.84\ psi)$, wing area and semi-span are 6050 in^2 and 110 inches. The wing skin consists of a symmetrical composite laminate, modelled as a plate with six symmetrical layers, $[\theta/+45°/-45°]_s$. The thickness of the each layer is constant and is not subject to change.
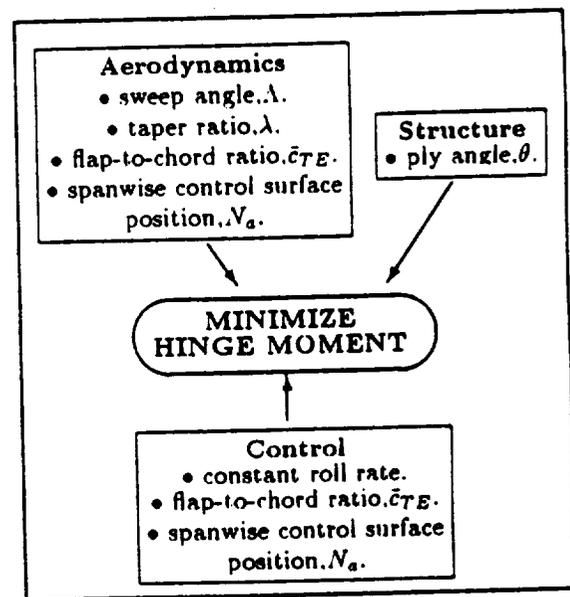
Figure 2: Wing design synthesis(first example).

Five design variables(listed in Figure 2) are used in this optimization example. The integrated optimal design system(shown in Figure 2) consists of three sub-systems(aerodynamics, structure, and control sizing and placement), which are the functions of the design variables. For the overall optimal design, three sub-systems must be accessed simultaneously. Although geometry

426

such as wing sweep and taper will change during optimization, wing area and span are fixed during the optimization procedure to keep the aspect ratio of the wing constant. Spanwise extent of the aileron is set to be 30% of the semi-span, but the chord dimension can change.

Figure 4 summarizes the changes in wing planform geometry and ply orientation during the optimal design process. The initial ply orientation θ(shown in Figure 1) is 5° while the wing quarter-chord sweep angle is 15°. These design variables converge to 32° and 25.5°, respectively. The optimal aileron position and flap-to-chord ratio are 84% of the semi-span and 0.2 which is the constraint boundary. Wing tip twist approaches the upper limit of the constraint as shown Figure 5. The optimal hinge moment is 884 lb-in.

In this optimization example, the aileron spanwise position was determined subject to the minimization of hinge moment only. The aileron also can be used for flutter suppression. Nissim's aerodynamic energy concept[8] was used to determine the most effective aileron spanwise location for flutter suppression. This result is compared with that for roll control.

Figure 6 shows the variation of "energy ratio" with the aileron spanwise location when the ply orientation is 45° and the quarter-chord sweep angle is 20°. The flutter speed V_f is required to be 888 feet per second when the control is active. A location corresponding to the larger negative value of energy ratio represents the more effective aileron spanwise location to suppress flutter. The energy ratio is computed when $V = 1.1V_f$. Figure 6 shows that the aileron should be placed at about 70% of the semi-span in this case. The best position of the aileron for roll control is also good for flutter suppression in this case, which is typical of others examined.

For the previous optimization example, wing weight was fixed. As the second optimization example, wing weight is included as a part of the objective function and the thickness coefficients of a skin layer are included as design variables.

The skin layer thickness distribution $t_i(\xi, \eta)$ is expressed as a power series:

$$t_i(\xi, \eta) = \sum_{m=0}^{t_m} \sum_{n=0}^{t_n} t_{mn} \xi^m \eta^n \qquad (4)$$

where ξ, η are the nondimensionalized coordinates such that ξ refers to the chord and η refers to the span, as shown in Figure 1. Nine thickness coefficients t_{mn} of the θ layer are included as design variables. The thickness distribution of the outer layer is written as follows:

$$t_1(\xi, \eta) = \sum_{m=0}^{2} \sum_{n=0}^{2} t_{mn} \xi^m \eta^n \quad inches \qquad (5)$$

The thickness distributions of the other two +45°, −45° layers are prescribed as follows:

$$t_2 = t_3 = 0.025 - 0.015\eta \quad inches \qquad (6)$$

The wing weight portion of the objective function is computed as follows:

$$W = \sum_{i=1}^{6} \int \int \rho t_i(\xi, \eta) d\xi d\eta \qquad (7)$$

where ρ is the mass density of the material and the number of material layers is 6.

A total of 12 design variables were used in the second example. These are shown in Figure 3.
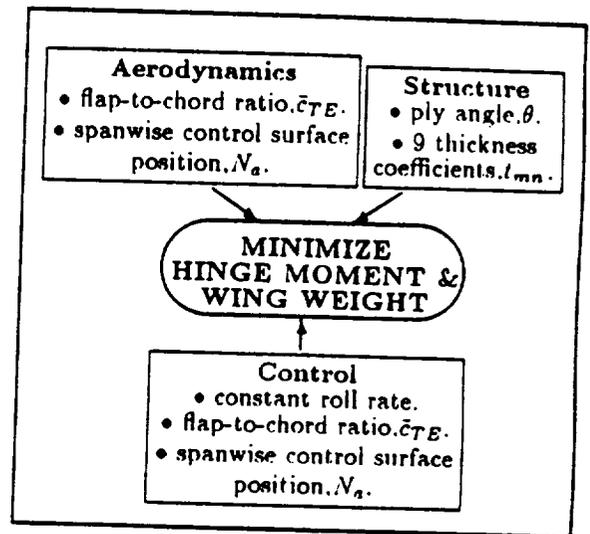
Figure 3: Wing design synthesis(second example).

Constraints for this optimization study are as follows:

1. $(pb/2V) = 0.12$

2. |wing tip twist| $\leq 3°$

3. $15\% \leq N_a \leq 85\%$ of the semi-span

4. $0.2 \leq \bar{c}_{TE} \leq 0.4$

5. Minimum thickness constraint (The layer thickness should be greater than zero.); The layer thickness is checked at 6 different locations (the leading and trailing edges of the wing root, mid-span, and tip).

The objective function J for the simultaneous minimization of the wing weight and aileron hinge moment is expressed as a linear combination of the wing weight and control surface hinge moment. This objective function is shown as follows:

$$J = \frac{W}{W_0} + \beta \frac{H}{H_0} \qquad (8)$$

427

where β is the weighting factor. This factor will depend upon the relative weighting placed upon structural weight and actuator weight. W and H are the wing weight and aileron hinge moment. W_0 and H_0 are the wing weight and aileron hinge moment for the initial set of the design variables. The aileron hinge moment H is a function of the thickness coefficients, ply orientation, and aileron spanwise location and chordwise dimension.

Figure 7 shows the optimized wing thickness distribution when the weighting factor β is 1.0. The thickness of the laminate is the largest at the leading edge near the wing root and the thickness constraint at the trailing edge of wing tip becomes active as the optimization progresses. The initial wing weight is 64.7lbs. The final wing weight and hinge moment are 50.2lbs and 1639.7 lb-in. The optimal ply orientation is 25° and the aileron mid-span location is about 75% of the semi-span.

Other examples from Reference [9] show that the optimization result is sensitive to the initial design variables. If the design process is started using a wash-in laminate design($0° \leq \theta \leq 90°$), the optimal ply orientation remains in the wash-in region and the wing sweeps aft to compensate for the effects of the wash-in laminate. The converse is true for the wash-out laminate($-90° \leq \theta \leq 0°$). To determine whether or not the minimized hinge moment is global, the optimization analysis must be conducted using other initial values of the design variables.

Conclusion

From these limited examples, we have seen that aileron size and location, structural arrangement and aerodynamic planform are highly interdependent. As a result, a parallel synthesis procedure is invaluable.

The aileron hinge moment can be minimized by redesigning the wing geometry and aileron flap-to-chord ratio and spanwise location, and reorienting composite ply angles. The hinge moment and wing weight can be minimized simultaneously. For the wing planform geometry used in this study, the final wing thickness distribution shows that the leading edge area of wing root is the thickest. The wing thickness decreases gradually in the direction of the trailing edge of wing tip.

Although no attempt was made to include flutter suppression or other maneuver criteria simultaneously, there was little difference between the best aileron location for roll control and that for flutter suppression.

References

[1] Weisshaar, T.A., "Aeroelastic Tailoring - Creative Uses of Unusual Materials," AIAA Paper No. 87-0976,AIAA/ASME/ASCE/AHS 28th Structures, Structural Dynamics and Materials Conferences, Monterey,California,April 1987.

[2] Sobieszanski-Sobieski, J., "Sensitivity Analysis and Multidisciplinary Optimization for Aircraft Design: Recent Advances and Results," 16th Congress of the International Council of the Aeronautical Sciences(ICAS), Jerusalem, Israel, 1988.

[3] Zieler, T.A., Weisshaar, T.A., "Integrated Aeroservoelastic Tailoring of Lifting Surfaces." Journal of Aircraft, Vol.25, No.1, January 1988.

[4] Weisshaar, T.A., Nam, C., "Aeroservoelastic Tailoring for Lateral Control Enhancement," Journal of Guidance, Control, and Dynamics, Vol.13, No.3, May-June 1990.

[5] Livne, E., Schmit, L.A., Friedmann, P.P., "An integrated Approach to the Optimum Design of Actively Controlled Composite Wings," AIAA Paper No. 89-1268,AIAA/ASME/ASCE/AHS/ASC 30th Structures, Structural Dynamics and Materials Conferences, Mobile, Alabama, April 1989.

[6] Giles, G.L., "Equivalent Plate Analysis of Aircraft Wing Box Structures with General Planform Geometry," NASA TM-87697, March 1986.

[7] Giles, G.L., "Further Generalization of an Equivalent Plate Representation for Aircraft Structural Analysis," NASA TM-89105, February 1987.

[8] Nissim, E., Burken, J.J., "Control Surface spanwise placement in Active Flutter Suppression Systems," NASA TP 2873, November 1988.

[9] Nam, C., "Aeroservoelastic Tailoring for Lateral Control Enhancement," Ph.D. Thesis, School of Aeronautics & Astronautics, Purdue University, May 1990.

[10] Avriel, M., Nonlinear Programming - Analysis and Methods, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1976.
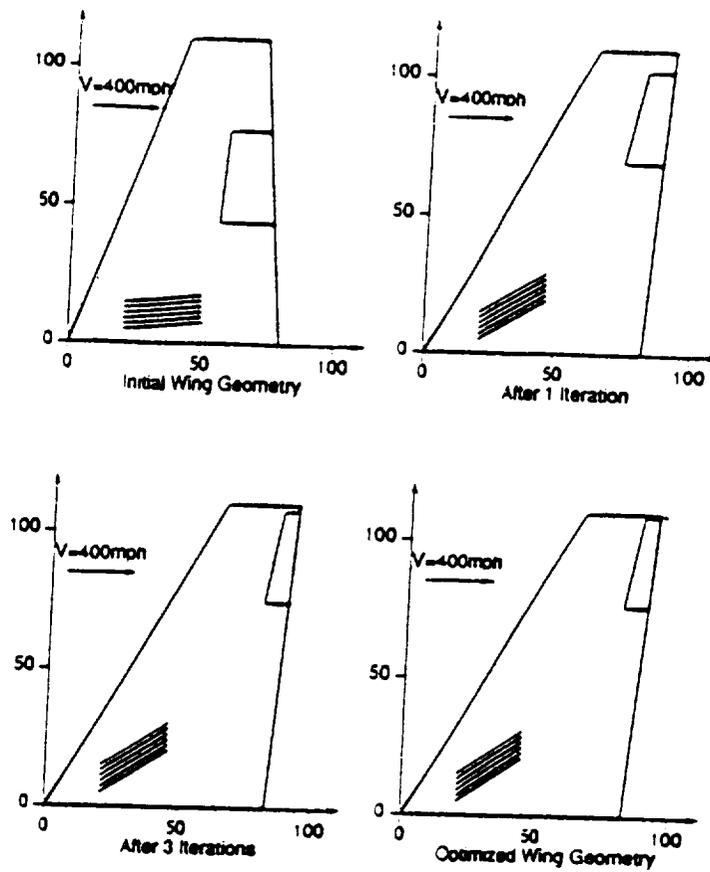
Figure 4: Changes in design variables during the optimization process.
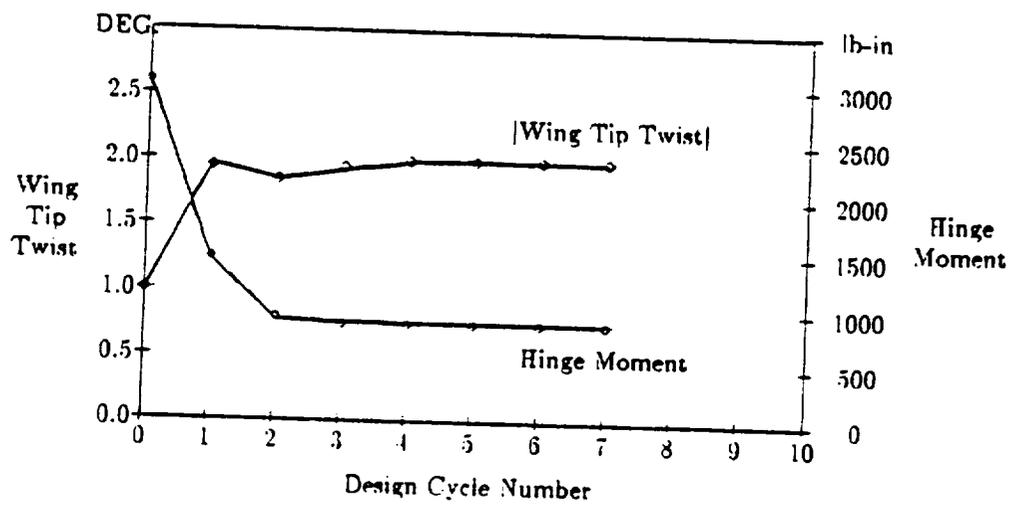
Figure 5: Wing tip twist and hinge moment versus design cycle number.

429
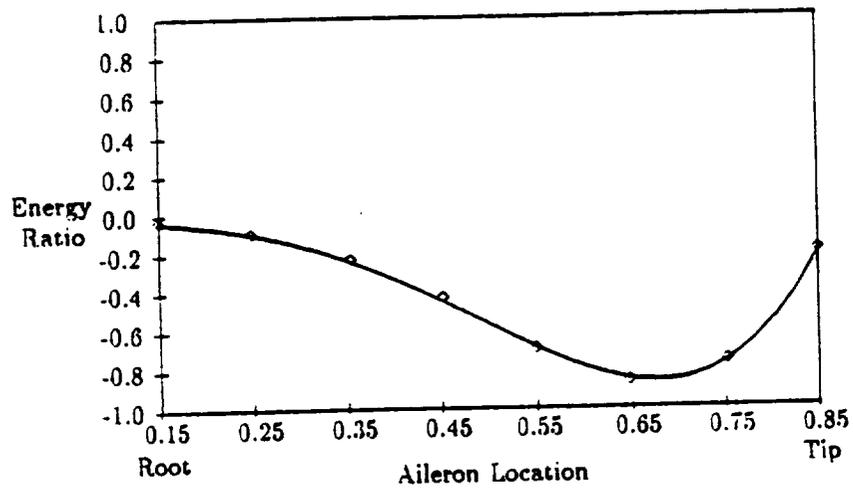
Figure 6: Energy ratio versus aileron location, when $\theta = 45°$.
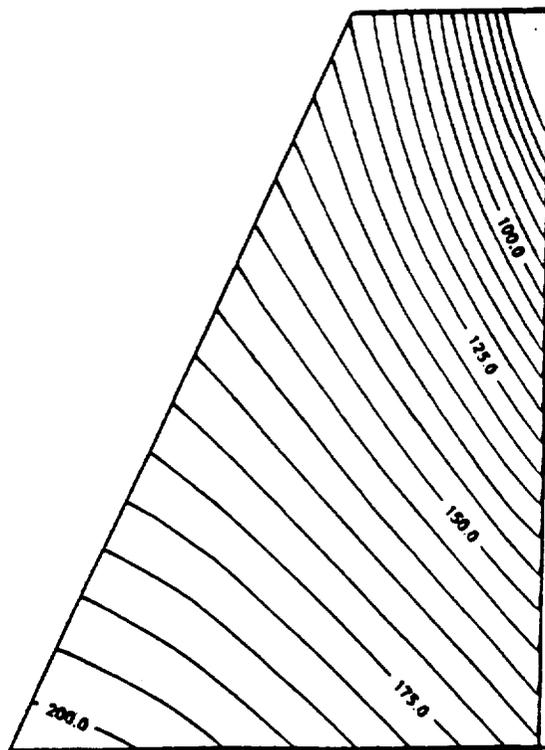
Figure 7: The optimized wing thickness(inches) distribution (multiplied by 1000) with weighting factor $\beta = 1.0$.

INFLUENCE OF STRUCTURAL AND AERODYNAMIC MODELING
ON OPTIMIZATION WITH FLUTTER CONSTRAINT

Alfred G. Striz
University of Oklahoma
Norman, Oklahoma 73019

Vipperla B. Venkayya
Wright Research and Development Center
Wright-Patterson AFB, Ohio 45433

Abstract

The influences of structural and aerodynamic modeling on multidisciplinary optimization in an aeroelastic environment are not yet well understood. Therefore, optimizations with flutter and frequency constraints were performed to investigate the effects these modeling factors have on various representative wings. To this end, the Automated STRuctural Optimization System (ASTROS) was used as a tool to minimize the weight of various fully built-up finite element wing models in subsonic and supersonic flow under given flutter and frequency constraints. First, the performance of the optimization module was tested against results from other codes on a straight and uniform wing widely used for optimization with flutter constraints. Then, fully built-up finite element models of various wings with different aspect ratios were investigated for the influence on the structural optimization for minimum weight of such modeling factors as finite element selection, structural grid refinement; number of selected modes, retention of breathing modes; selection of reduced frequencies to be used in flutter analysis; aerodynamic panel size and placement; splining of the aerodynamic grid to the structural grid, selection of extra points off the structural wing box for splining; number of constraints to be retained; etc. Knowledge of these influences as well as of the program behavior is important, since optimization can be made more efficient by the selection of reasonable initial models. Also, it was shown previously that modeling has an impact on the results of modal and aeroelastic analyses. Thus, if modeling errors can negatively affect the analyses, a minimum weight optimization can be jeopardized and result in an optimal design that is rather unreliable. In the following, selected results are presented and the influences of some modeling parameters on optimization are pointed out.

1. Introduction

In recent years, structural optimization as required and applied by the aerospace industry has expanded in scope to include such additional disciplines as static and dynamic aeroelasticity, composite materials, aeroelastic tailoring, etc. One of the more promising multidisciplinary codes presently under development is the Automated STRuctural Optimization System (ASTROS) [1-3]. In this computer code, static, dynamic, and frequency response finite element structural modules, subsonic and supersonic steady and unsteady aerodynamic modules, and an optimization module are combined and allow for either analysis or optimized design of given aircraft configurations. Interfering surface aerodynamics are incorporated to handle the aerodynamic modeling of combinations of wings, tails, canards, fuselages, and stores. Structures are represented by finite element models, constructed from rod, membrane, shear, plate, and other elements. Static and dynamic aeroelastic capabilities include trim, lift effectiveness, aileron effectiveness, gust response, and flutter analysis.

In the present paper, as part of an ongoing effort to gain a better understanding of the optimization process with aeroelastic constraints, the optimization portion of ASTROS was used together with the normal modes and flutter module for various investigations of fully built-up finite element wing models in subsonic and supersonic flow to determine the influences of structural and aerodynamic modeling as well as splining on optimization with flutter constraints and, thus, to investigate the behavior of the flutter and optimization modules of the code. This knowledge is incidental to the understanding of the dynamic

behavior of wings during the optimization process. It is also expected to result in better initial models and, thus, a more efficient optimization cycle.

First, the performance of the optimization module was evaluated against results reported in the literature on the straight untapered wing used by Rudisill and Bhatia [4], McIntosh and Ashley [5], Segenreich and McIntosh [6], and others for structural optimization with flutter constraints. This wing was chosen since it represents one of the very few models where all structural, material, and environmental data are given for aeroelastic analysis and optimization with flutter constraints. Then, a set of test cases was selected consisting of a high aspect ratio swept and tapered wing, a medium aspect ratio straight wing with a tapered section toward the wing tip, and a low aspect ratio swept and tapered fighter-type wing (Fig. 1). The straight wing and the high aspect ratio wing were evaluated at subsonic Mach numbers while the fighter wing was investigated for flutter at subsonic and supersonic speeds. These latter three wings were modified derivatives of the wings used in the investigation of the influence of modeling on normal modes and flutter analysis by Striz and Venkayya [7].

High Aspect Ratio Wing   Medium Aspect Ratio Wing   Low Aspect Ratio Wing
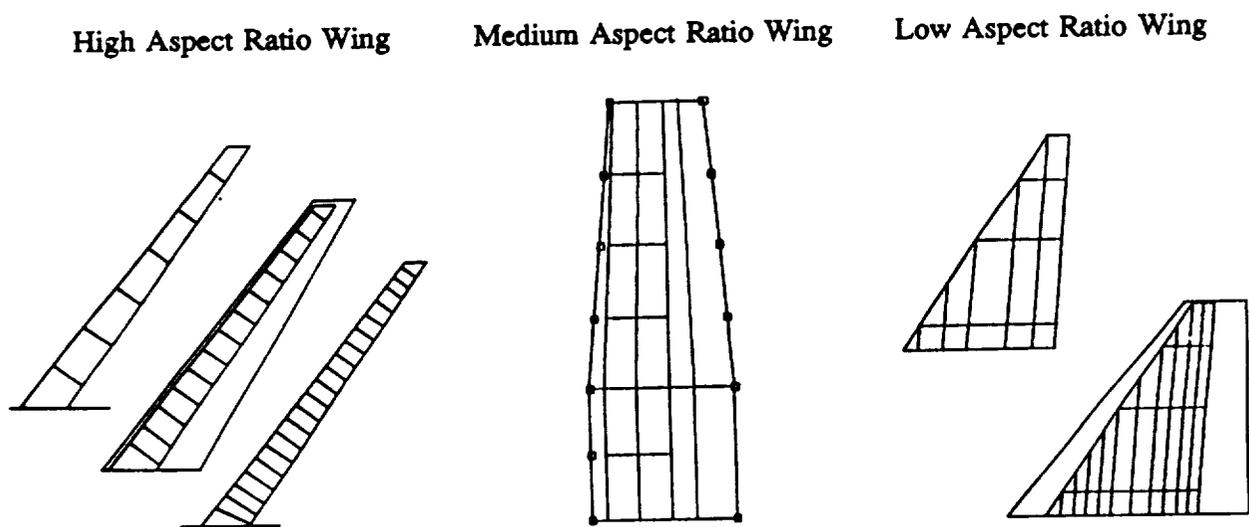
Figure 1. Wing Models Used in the Optimization Test Cases

2. Background

The importance of this investigation can be seen from the following example: It is generally understood that membrane elements when used for spars and ribs overpredict the stiffness of a wing. Thus, when the wing used by Rudisill and Bhatia was modeled by the present authors by replacing the front and rear spar membrane elements with shear elements, the natural frequencies of the first three bending modes dropped from 10.5, 55.9, and 125.8 Hz to 6.3, 37.6, and 110.3 Hz, respectively. This kind of change in wing bending frequencies can have a considerable impact on control surface performance and flutter. However, this example represents only a structural modeling change. In flutter analysis and optimization with aeroelastic constraints, the aerodynamic modeling also affects the results: the number, size, and distribution of the aerodynamic panels and the splining between the aerodynamic points and the structural grid. Since optimization is only as good as the associated analyses, it can, in some cases, compound and exaggerate errors arising from these. Thus, if modeling errors can have a considerable impact on the quality of the results of the associated analyses [7], optimization can be seriously jeopardized to the point where the resulting optimal design can be very unreliable. In the cited example, use of the stiffer membrane elements resulted in a 10% lower minimum weight design (38 lbs) as compared to the more

realistic, less stiff shear elements (42 lbs). If flutter is the driving constraint, this could lead to the design of a structure that is potentially too weak. It is, therefore, essential that the initial designs used in optimization are feasible and modeled correctly especially when built-up finite element structural models are used rather than the previously more common beam models.

Thus, fully built-up finite element structural models for the four wings were evaluated for their behavior in optimization and the influence of such modeling factors as finite element selection, structural grid refinement; selection of upper frequency bounds; aerodynamic panel size and placement; splining of the aerodynamic grid to the structural grid; the selection of extra points off the structural wing box (MPCs) for better mass distribution and aerodynamic splining; solution procedures such as reduction schemes; etc., and selected results are presented.

3. The Rudisill and Bhatia Wing Model

As mentioned in Reference 7, one of the drawbacks of this model lies in the fact that no non-structural mass was included in the model. Thus, the mass center of the wing coincides with the elastic axis, resulting in a close proximity of flutter speed and divergence speed. Here, for the base model with skins, ribs, and webs all modeled by membrane elements, the flutter speed for an input Mach number of M = 0.5566 and an altitude of h = 10,000 ft was calculated by ASTROS and MSC/NASTRAN as 10,881 in/sec and 10,500 in/sec, respectively, with divergence speeds of 11,900 in/sec and 11,500 in/sec, respectively. The flutter speed shown in Fig. 3 of Reference 4 for the initial configuration was about 10,800 in/sec. When the optimized versions of the model as obtained in References 5 and 6 were analyzed for flutter, they were found to all encounter a divergence speed much lower than the speed used as a flutter constraint. It seems that none of these optimizations included the possibility of divergence as a flutter root with zero frequency. Thus, these optimized results seem to have been limited to flutter constraints only and would have resulted in designed wing models that considerably exceeded their divergence speeds.

First, in order to test the influence of the finite element selection on the optimization, the spar webs as well as the ribs were alternately modeled as shear elements and as membrane elements. The rest of the model was kept as in Reference 4. All inplane displacements were removed from the analysis set by Guyan reduction. Optimization results are presented in Table 1.

It can be seen that changing the ribs from membrane elements to shear elements did not influence the optimized weight at all. However, when the spar webs were changed from membranes to the more realistic shear elements, there was a significant increase in the optimized weight due to the fact that the natural frequencies, especially for the bending modes, as well as the divergence and flutter speeds all dropped significantly, showing the all-membrane model to be non-conservative. When non-structural masses were added to the all-shear model, the minimum weight stayed essentially the same, but now

Table 1. Varying Element Types on Wing of Reference 4 (9 Design Variables)

Rib Elements: Spar Elements:	Membrane Membrane	Shear Membrane	Membrane Shear	Shear Shear	Shear/Mass Shear/Mass
Initial Structural Weight: Optimum Structural Weight: Aeroelastic Mode:	195.92 37.69 Divergence No Flutter	195.92 37.69 Divergence No Flutter	195.92 41.76 Divergence Flut. Close	196.04 41.79 Divergence Flut. Close	196.04 41.68 Divergence Flutter

the divergence and flutter speeds almost coincided for the optimized structure.

To examine the influence of the number of aerodynamic boxes on the wing, various paneling schemes were chosen for the model with shear elements for spar webs: 6 spanwise boxes x 4 chordwise boxes, 6 x 9, 24 x 4, and 24 x 9, respectively. The results suggest that a reasonably coarse mesh, especially in chordwise direction, can be used to save computer time for preliminary optimization and design, since it seems to result in a conservative approximation to the minimum weight.

4. Three Wing Models with Different Aspect Ratios

The three wing models represent, in that order, a swept and tapered transport/bomber type wing of high aspect ratio, a straight and partially tapered light transport/combat aircraft type wing of medium aspect ratio, and a swept and tapered fighter type wing of low aspect ratio. All necessary dimensions and parameters are available in Reference 8. Some selected structural and environmental data for these wings are given in Table 2.

The structural models for the three wings were built from rod, membrane, and shear elements to

Table 2. Environmental, Initial Geometrical, and Material Property Model Data

HIGH ASPECT RATIO WING: (Transport/Bomber, M = 0.87, h = 30,000 ft)

Variation:	Seven ribs, fourteen ribs, twenty-one ribs	
Thick-	Shear panels:	0.145" to 0.1" in ribs (for 14-rib); 0.2" to 0.1" in spars
nesses:	Membranes:	0.3" to 0.1" in skins
Areas:	Spar stiffeners:	0.15 in² (for 14-rib) Spar caps: 3.6 to 3.0 in²

MEDIUM ASPECT RATIO WING: (Light Transport/Combat Aircraft, M = 0.58, h = 5,000 ft)

Variations:	No MPCs, aerodynamic MPCs, mass MPCs; aerodynamic mesh variations			
Thicknesses:	Shear panels:	0.08" in spars/ribs	Membranes:	0.06" in skins, 0.08" in ribs
Areas:	Spar stiffeners:	0.2 in²	Spar Caps:	1.0 in²

LOW ASPECT RATIO WING: (Fighter, M = 0.85, h = 5,000 ft; M = 1.5, h = 30,000 ft)

Variation:	Five spars, ten spars	
Thick-	Shear panels:	0.08"{I} / 0.12"{II} in ribs; 0.15 to 0.06" in spars (5-spar)
nesses:		0.135 to 0.05" le/te, 0.075 to 0.03" int., in spars (10-spar)
	Membranes:	0.25 to 0.04" in skins
Areas:	Spar caps:	2.0 to 1.0 in² {I} / 1.0 to 0.5 in² {II} (5-spar)
		1.75 to 0.88 in² le/te, 1.0 to 0.5 in² int. (10-spar)
	Spar stiffeners:	0.05 in²

Material for all wings is Aluminum: E = 10,000,000 lb/in², ν = 033, ϱ = 0.1 lb/in³.
All values decreasing from root to tip.

Table 3. Spanwise Structural Variation, High Aspect Ratio Wing (7 X 5 Aero Mesh)

# of Ribs: # of Design Variables:	Seven		Fourteen		Twenty-One	
	13	26	13	26	13	26
Initial Structural Weight:	10205.6	10205.6	10205.4	10205.4	10205.2	10205.2
Optimum Structural Weight:	6408.8	6341.3	6498.3	6447.8	6371.9	6352.4

represent the wing boxes with spars, spar caps, spar stiffeners, ribs, and skins. Here, the rods corresponded to spar caps and spar stiffeners, the membranes were used for the skins, and the shear elements for the spar webs and the ribs of the wings.

a) <u>High Aspect Ratio Wing</u>: For the high aspect ratio wing, the structural weight was assumed to be about 30% of the overall weight of the wing, with the other 70% distributed as non-structural masses at all nodal points. In the optimization process, Guyan reduction was applied to retain out-of-plane displacements only. A flutter constraint of 14,000 in/sec was chosen together with a lower bound of 1 Hz on the lowest natural frequency. For this wing, the influence of structural complexity in spanwise direction was evaluated by varying the number of bays in the wing from seven to fourteen to twenty-one while keeping the total weight constant. This increase accounts for a more uniform distribution of mass and stiffness without changing their overall values. Reasonable element aspect ratios were exceeded to determine how forgiving the structural modeling process is (Table 3).

From these results, it seems that the most reasonable fourteen bay wing showed the most conservative results while the other two wings yielded lower minimum weights. This could be due to the stiffness distributions in the respective models, especially in the root area, or due to the somewhat excessive aspect ratios. In all cases, an increase in the number of design variables resulted in a lower weight as expected since a finer discrete distribution of masses is possible.

b) <u>Medium Aspect Ratio Wing</u>: For all models of the medium aspect ratio wing, the structural weight was assumed to constitute about 30% of the overall weight of the wing, with the other 70% distributed as non-structural masses at all structural nodal points and MPCs. The flutter constraint chosen was 14,000 in/sec. Here, the influence of the aerodynamic wing model complexity was evaluated. The number of aerodynamic boxes on the wing was increased from an initially very coarse grid (5 spanwise boxes by 5 chordwise boxes) by increasing the number of spanwise and chordwise subdivisions. The reasonable aspect ratio of the aerodynamic boxes was exceeded to determine how forgiving the aerodynamic modeling process is. The results for some of these cases are presented in Table 4.

Here, the models with less spanwise boxes showed slightly higher minimum weights with virtually no variation due to a change in the number of chordwise boxes. This seems to indicate that a coarse aerodynamic mesh can be used for preliminary design and will result in a conservative design.

Table 4. Aerodynamic Mesh Variation, Medium Aspect Ratio Wing (31 Design Variables)

Panel Mesh:	5 x 5	5 x 10	11 x 5	11 x 10	22 x 5	22 x 10
Initial Structural Weight:			576.8			
Optimum Structural Weight:	177.7	177.3	170.6	168.6	167.5	166.5

Table 5. Use of MPCs, Medium Aspect Ratio Wing, I: EPS=-0.02; II: EPS=-0.03; III: EPS=-0.05
(5 x 5 Aero Mesh, 31 Design Variables)

MPCs: Upper Frequency Bound:	None 50 Hz	 100 Hz	Aero 50 Hz	 100 Hz	Mass 50 Hz	 100 Hz	Aero+Mass 50 Hz	 100 Hz
Init. Struct. Weight:				576.8				
Opt. Struct. Weight: I	170.3	184.2	157.4	157.1	229.9	477.0	175.6	180.0
II	179.1	184.2	157.4	157.1	229.9	477.0	175.3	175.6
III	179.1	186.4	157.4	157.1	229.9	477.0	175.6	206.4

Finally, the use of multi-point constraints (MPCs) was evaluated. These MPCs add non-structural points rigidly splined to existing structural points for two purposes: to attach non-structural masses for better overall mass distribution and to add points to which the aerodynamic loads can be splined for better aerodynamic load distribution. They had been used in all of the above mentioned computations for the medium aspect ratio wing along with Guyan reduction to only out-of-plane displacements. Three different values of the constraint retention parameter EPS were applied: -0.02, -0.03, and -0.05, as well as two values for the upper frequency bound on the modal flutter analyses: 50 Hz and 100 Hz. For this study only, the vertical spar stiffeners were eliminated and the ribs converted from shear to membrane elements to eliminate breathing modes.

It seems (Table 5) that, for a given combination of upper frequency limit and constraint retention parameter EPS, the use of MPCs for better distribution of the non-structural mass away from just the structural wing box has the effect of increasing the optimized weight coupled with a lowering of the flutter speed found in the accompanying analysis. This may be caused by the larger rotational moments produced by these offsets. The use of MPCs for splining the aerodynamic forces to a larger area than just the structural wing box had the opposite effect, i.e. the optimized weight was even lower than for the case with no MPCs. This was consistent with a considerable increase in the flutter speed from the accompanying analysis. When the two sets of MPCs were combined, however, the minimum weight of the structure was comparable to that for the case of no MPCs. Thus, mass MPCs seem to be a necessity for obtaining a conservative weight in optimization, even though the lack of aerodynamic MPCs may result in too high a minimum weight. Also, an increase in the upper frequency limit, i.e. in the number of modes retained in the flutter analyses, resulted in an increase in the minimum weight for all but the aerodynamic splining results while the effect of a change in the constraint retention parameter had, for most cases, little influence. However, both of these parameters have to be chosen with care.

c) Low Aspect Ratio Wing: For the low aspect ratio wing, non-structural mass in the amount of 2400 lbs was distributed over all nodal points and a mass of 200 lbs for a tip store with launcher over the wing tip points. No MPCs were used, since the wing box covers a large part of the projected wing area. Aerodynamic meshes of 5 x 5 and 15 x 15 boxes were chosen. A flutter constraint of 25,000 in/sec was applied. For this wing, the influence of structural complexity in chordwise direction was evaluated. Starting with a reasonable model for the wing box using five internal spars, the wing was subdivided by adding five more spars while keeping the total weight constant. The influence of a more evenly distributed stiffness and mass arrangement was thus evaluated. Both, subsonic and supersonic analyses were performed. Results for the subsonic case only (M = 0.85) are presented in Table 6 for aerodynamic meshes of 5 x 5 and 15 x 15 boxes.

The results suggest that distributing mass and stiffness more evenly in chordwise direction allows

Table 6. Varying Spar Number on Low Aspect Ratio Wing, Aero Mesh a) 5 x 5 b) 15 x 15

# of Internal Spars:		Five			Ten	
# of Design Variables:		6	18		6	26
Initial Structural Weight:	I	497.8			497.7	
	II	402.7				
Optimum Structural Weight:	Ia	330.3	228.0	Ia	303.6	202.8
	Ib	352.6	237.0	Ib	328.5	208.6
	IIa	322.6	218.6			
	IIb	362.4	228.4			

the optimization to optimize more members and thus leads to lower final weights. The same is, of course, true when the number of design variables is increased. It should be noted that the five spar wing with 18 design variables resulted in a lower weight than the ten spar wing with six design variables suggesting that it might be advantageous for the preliminary sizing of wings with flutter constraints to use a relatively simple model with a reasonably large number of design variables rather than go through the effort of creating a more complex model. Since the initial structure {I} of the five spar wing had somewhat oversized spar caps but undersized shear webs, both sets of values were adjusted in structure {II} to result in a 19.1% lighter wing with a more balanced size and mass distribution. However, this only resulted in a slightly lower overall weight in the optimization (less than 5% for the structural weight and less than 0.5% for the total weight of the wing). When the fine aerodynamic mesh was chosen (15 x 15) rather than the coarse (5 x 5), the resulting minimum weights were somewhat higher (generally less than 12% for the structural weight and less than 1.5% for the total weight of the wing). However, for preliminary sizing, the coarser mesh resulted in much shorter CPU times (for ten spar wing with 26 design variables, the CPU times were 0:12:06 for the 5 x 5 mesh and 1:28:55 for the 15 x 15 mesh on the WRDC/FDL VAX8650).

5. Discussions and Recommendations

The influences of structural and aerodynamic modeling on optimization and the minimum weight design of built-up finite element wing models were investigated using the optimization module of the Automated STRuctural Optimization System (ASTROS). This was done to gain a better understanding of the optimization process with dynamic aeroelastic, i.e. flutter, constraints. Several trends could be observed during the course of the modeling and the optimization even though it is understood that, until many more cases have been evaluated, any set of analyses has to be regarded as more or less wing type and model specific.

A quick initial evaluation of a preliminary design with a reasonably coarse model for both the structure and the aerodynamics will result in an optimum weight that is close to that for a more detailed model though not always conservative. Here, a finer chordwise structural distribution seems to yield a better pay-off in terms of a lower minimum weight while, for the aerodynamic modeling, a finer spanwise distribution seems preferable.

The selection of the correct finite elements for modeling the structure is rather critical since, e.g., choosing membrane instead of shear elements for spars can result in non-conservative minimum weights in optimization. Further, care has to be taken when selecting the modes included in the optimization. In-plane modes as well as extensional modes of the vertical spar connecting rods can cause convergence problems and should be eliminated. For wings where chordwise bending modes are not expected, it is

suggested to increase the frequency of the extensional modes by eliminating the connecting rods and converting the shear elements generally used for ribs to membrane elements. For fighter type wings with possible chordwise bending modes, the upper and lower wing surfaces can be connected by MPCs instead. Finally, the number of modes retained for modal flutter analysis during the course of an optimization can affect the computed optimum weights as can the selection of the constraint retention parameter. Thus, these two parameters have to be carefully chosen.

The use of mass MPCs is advised for a more realistic mass distribution and seems to yield rather conservative results, while the use of aerodynamic MPCs for a better aerodynamic force distribution seems to lower the minimum weights in a non-conservative fashion. Depending on the model, the omission of all MPCs can lead to lower weights and, thus, be non-conservative as well.

Future work will include investigations into the influence of how the aerodynamic forces are splined to the structure, into the effect of input Mach number on optimized weight, and into the use of move limits in optimization. Optimization with strength and static aeroelastic constraints is being performed at present. Eventually, these constraints will be combined with the flutter constraint to evaluate the behavior of representative wings in a true multi-disciplinary optimization environment and to allow for a more general understanding of the modeling influences on such optimization.

Acknowledgments

References

1. Johnson, E.H., and Venkayya, V.B., "Automated Structural Optimization System (ASTROS), Vol. I: Theoretical Manual", AFWAL-TR-88-3028/I, Air Force Wright Aeronautical Laboratories, December 1988.

2. Neill, D.J., Johnson, E.H., and Herendeen, D.L., "Automated Structural Optimization System (ASTROS), Vol. II: User's Manual", AFWAL-TR-88-3028/II, Air Force Wright Aeronautical Laboratories, April 1988.

3. Johnson, E.H., and Neill, D.J., "Automated Structural Optimization System (ASTROS), Vol. III: Applications Manual", AFWAL-TR-88-3028/III, Air Force Wright Aeronautical Laboratories, December 1988.

4. Rudisill, C.S., and Bhatia, K.G., "Optimization of Complex Structures to Satisfy Flutter Requirements", AIAA Journal, Vol. 9, No. 8, August 1971, pp. 1487-1491.

5. McIntosh, S.C. Jr., and Ashley, H., "On the Optimization of Discrete Structures with Aeroelastic Constraints", Computers and Structures, Vol. 8, No. 3/4, 1978, pg. 411-419.

6. Segenreich, S.A., and McIntosh, S.C., "Weight Minimization of Structures for Fixed Flutter Speed Via an Optimality Criterion", AIAA-75-0779, AIAA/ASME/SAE 16th Structures, Structural Dynamics, and Materials Conference, Denver, Colorado, May 27-29, 1975.

7. Striz, A.G., and Venkayya, V.B., "Influence of Structural and Aerodynamic Modeling on Flutter Analysis", AIAA-90-0954-CP, Proceedings, 31st AIAA/ASME/ASCE/AHS/ACS Structures, Structural Dynamics and Materials Conference, Long Beach, California, April 1990, pp. 110-118.

8. Striz, A.G., "Realistic Structural and Aerodynamic Wing Models for Flutter Analysis and Structural Optimization", OU-AME-TR-90-110, University of Oklahoma, Norman, Oklahoma, January 1990 (updated September 1990).

Preliminary Design Optimization of Joined-Wing Aircraft

John W. Gallman,* Ilan M. Kroo,† and Stephen C. Smith *

Introduction

The joined wing is an innovative aircraft configuration that has a its tail connected to the wing forming a diamond shape in both top and plan view.[1] This geometric arrangement utilizes the tail for both pitch control and as a structural support for the wing. Several researchers have studied this configuration and predicted significant reductions in trimmed drag or structural weight when compared with a conventional T-tail configuration.[2,3] Kroo et al.[4] compared the cruise drag of joined wings with conventional designs of the same lifting-surface area and structural weight. This study showed an 11% reduction in cruise drag for the lifting system of a joined wing. Although this reduction in cruise drag is significant, a complete design study is needed before any economic savings can be claimed for a joined-wing transport. Mission constraints, such as runway length, could increase the wing area and eliminate potential drag savings.

Since other design codes do not accurately represent the interaction between structures and aerodynamics for joined wings, we developed a new design code for this study. The aerodynamics and structures analyses in this study are significantly more sophisticated than those used in most conventional design codes. This sophistication was needed to predict the aerodynamic interference between the wing and tail and the stresses in the truss-like structure. This paper describes these analysis methods, discusses some problems encountered when applying the numerical optimizer NPSOL,[5] and compares optimum joined wings with conventional aircraft on the basis of cruise drag, lifting surface weight, and direct operating cost (DOC).

Analysis Methods

The analysis methods required to estimate transport aircraft performance throughout all phases of a medium range mission were developed for the design code. However, the most interesting optimization problems were associated with the aerodynamic and structural analyses. Since these analyses are also the most important for the estimation of joined-wing performance, they are described below. Methods used to calculate mission constraints such as range and runway field length are described in ref. 6 along with the objective function, DOC.

The aerodynamic analysis is performed by a subroutine version of the LinAir[7] program. This program solves the Prantl-Glauert equation with a vortex-lattice model and calculates induced drag using Trefftz-plane integration. Figure 1 shows lift distributions for both the wing and tail of a typical joined wing and a top view of the vortex-lattice model. The aerodynamic modeling is sophisticated enough to represent changes in the wing's lift distribution due to flap deflection and changes in the tail's lift distribution due to elevator deflection. Fuselage and engine nacelle models were needed to accurately calculate the downwash on the tail and hence the tail's contribution to pitching moment and static stability.

Both the wing and tail structures are modeled with variable cross section skin-stringer beams (see Fig. 2). These beams utilize an asymmetric material distribution to support a bending moment which tends to be oriented diagonally across the structural box, as described in ref. 1. Joining the wing and tail structures with a rigid joint causes reaction forces that relieve wing bending moments inboard of the joint. Since a solution of beam deflections is required to calculate the magnitude of these joint reaction forces, the stress distribution depends on the stiffness of the structure. This dependency of stress on stiffness forces our design algorithm for this structure to be iterative.

A fully-stressed design method sized the structure with a closed-form solution for material thickness in terms of structural loads and the allowable stress. Multiple load cases including positive gusts, negative gusts, and maneuver loads are considered. The method analyzes the

---

* Aerospace Engineer, *NASA Ames Research Center, Moffett Field, California*
† Assistant Professor, *Stanford University, Stanford, California*

structure with an initial material distribution for each load case, calculates the thicknesses such that the maximum stress in all members is equal to the allowable stress and determines the structural weight of the wing and tail. These structural weights are compared with the weights associated with the initial material distribution. If the new and old weights differ by more that 10^{-5}, the method updates the initial distribution and repeats the analysis. A more detailed description of this method is given in ref. 6. Solving for member thicknesses with this method reduced the number of design variables for the optimization problem by 160, and hence the computer time required for an optimum solution.

Optimization

Two interesting problems were encountered during application of numerical optimization to the design problem. The first problem was associated with an error in the calculation of induced drag which was caused by a discontinuity in the panel width of the vortex-lattice model. This error caused the numerical optimizer to suggest that unrealistic wing twists were optimum. Figure 3 shows lift distributions calculated using a wing-fuselage model with constant panel widths and a model with a discontinuity in panel width at the wing-fuselage joint. Although the panel discontinuity is exaggerated in Fig. 3, a difference of 10% in panel width produced significant errors in the twist distributions of optimum solutions. Since a constant panel width was needed to obtain realistic solutions, wing and tail span had to be input parameters rather than continuous design variables. The second problem was function noise which was caused by the convergence criteria used in the structural design algorithm. This problem was solved by exercising the input option for NPSOL that accepts a user specified function precision. Using this option allowed an update of the initial material distribution without introducing a discontinuity in problem functions and saved 33% in overall computational time.

The optimization problem consisted of 11 design variables and 9 constraints. Fig. 4 shows the design variables, the constraints, and a flowchart of the design code. Explicit nonlinear constraints simplified the design code by eliminating the iteration loops found in other design codes.[8] NPSOL, a gradient-based optimizer, used a sequential quadratic programming algorithm to solve the design problem. This algorithm solves a quadratic subproblem to determine the appropriate search direction in design space and performs a line search in this direction which minimizes an augmented Lagrangian merit function. Figure 5 shows the iteration history of the objective function, and the norm of the constraint function violations for a typical 11 variable 9 constraint problem. This figure shows convergence to the Khun-Tucker conditions for optimallity in two hours and twenty minutes of computational time on a VAX 3200 workstation. Increasing the feasibility tolerance for the constraint violations from 10^{-5} to 10^{-3} could save much of this computational time during conceptual design.

Results

All joined-wing design results are compared with similarly designed conventional T-tail aircraft on the basis of drag, weight, and DOC. The values calculated for drag, weight, and DOC are all normalized with respect to values calculated for a reference airplane that is very similar to a MacDonnell Douglas DC-9-30 transport. Since the Pratt and Whitney JT8D-9 low by-pass ratio engine was used on the DC-9-30, a variable-size version of this engine is used throughout this design study. The mission considered is to transport 115 coach passengers 900 nautical miles at a cruise Mach number of 0.78. All designs have a stick-fixed static margin of 0.359 based on a reference chord of 13.69 ft. A balanced runway length of 6500 ft at sea level is used for the take-off and landing field length constraints. Other constraints include a second segment climb gradient greater that 2.4%, a cruise thrust greater than cruise drag, and cruise trim at cruise lift coefficient. Sufficient pitching moment for take-off rotation is also required.

Parametric studies of both wing and tail span were performed for joined-wing transports. These studies indicated that the optimum tail-to-wing span ratio was approximately 0.7, as suggested in ref. 4 for a single cruise design point. Also in agreement with ref. 4 is the trade-off between drag and lifting surface weight shown in Fig. 6 for joined wings with a tail-to-wing span ratio approximately equal to 0.7. Figure 6 shows joined wings with a span of 110 ft having approximately 11% less cruise drag than the reference airplane. The 99 ft span joined wing

indicates a savings of 8% in cruise drag and 10% in lifting surface weight. Reference 4 assumed that the joined wing would have a lifting surface weight equal to a conventional design to resolve the trade-off between drag and weight. This study indicates that the design with the lowest DOC (9% drag savings) has a relative weight of 0.95 instead of 1.0. A compromise between maintenance costs which increase with span (weight) and fuel costs which decrease with span (drag) produces this savings in DOC. Our studies have shown that considering a complete mission reduces the potential savings in DOC from 3.2% to the 2% shown in Fig. 6 for a joined wing with a span of 104.5 ft. The primary reason for this is that a larger wing was required to meet the take-off and climb constraints. This increase in surface area increases the skin friction drag which in turn increases the DOC. Extra wing area was needed for these joined-wing designs because of the larger tail downloads required to trim at the high wing lift coefficients used in second segment climb. Therefore, maximum lift capability seems to be a crucial design issue for joined wings.

A study of the sensitivity of these "optimum" designs with respect to the take-off, climb, and static stability constraints is needed before choosing the best joined-wing or conventional configuration. NPSOL chose a wing area of 825 ft^2 for the optimum conventional T-tail design which was 175 ft^2 less than the reference airplane. This could be caused by a difference in fuel costs or runway length constraints used to design the reference airplane. The optimum wing area for the joined wing was 925 ft^2. The large tail downloads that cause this increase in wing area also increase the induced drag of joined wings during cruise. Since reducing the required static margin will decrease the trim loads, further improvement can expected for joined wings with reduced stability. Conventional designs require very little tail load to trim indicating less potential for improvement with reduced stability.

Conclusions

The results presented demonstrate the usefulness of numerical optimization in preliminary aircraft design and indicate that joined wings are promising alternatives for medium range transports. A 9% savings in drag with a corresponding savings of 2% in DOC was shown for a joined-wing transport with a span of 104.5 ft and a tail-to-span ratio of approximately 0.7. A sensitivity study with respect to static stability, climb gradient, and runway length would also help identify the best joined-wing transport. Optimum solutions were determined efficiently and accurately by using constant panel widths in the aerodynamics model, decomposing the structural design from the optimization problem, and developing analysis methods for the design code that do not use iteration loops. Future application of these ideas should help optimization become a more useful design tool.

References

[1]Wolkovitch, J., "The Joined Wing: An Overview," *Journal of Aircraft*, Vol.23, March 1986, pp.161-178.

[2]Letcher, J., "V-Wings and Diamond-Ring Wings of Minimum Induced Drag," *Journal of Aircraft*, Aug. 1972.

[3]Miura, H., Shyu, A. and Wolkovitch, J., "Parametric Weight Evaluation of Joined Wings by Structural Optimization," *Journal of Aircraft*, Vol. 25, 1988, pp. 1142-1149.

[4]Kroo, I. M., Gallman, J. W. and Smith S. C., "Aerodynamic and Structural Studies of Joined-Wing Aircraft," AIAA Paper 87-2931, Sept. 1987, Submitted to Journal of Aircraft.

[5]Gill, P. E., Murray, W., Saunders M. A. and Wright M. H., "User's Guide for NPSOL (Version 4.0): A Fortran Package for Nonlinear Programming," Technical Report SOL 86-2, Department of Operations Research, Stanford University, Jan. 1986.

[6]Gallman, J. W., Kroo, I. M., and Smith, S. C., "Design Synthesis and Optimization of Joined-Wing Transports," AIAA Paper 90-3197, Sept. 1990.

[7]Kroo, I. M., "A Discrete Vortex Weissenger Method for Rapid Analysis of Liftng Surfaces," Desktop Aeronautics, P.O. Box 9937, Stanford CA, 94305, Aug. 1987.

[8]De Filippo, R., "Aircraft Synthesis using Numerical Optimization Methodology."," AIAA paper 83-2458, Oct. 1983.
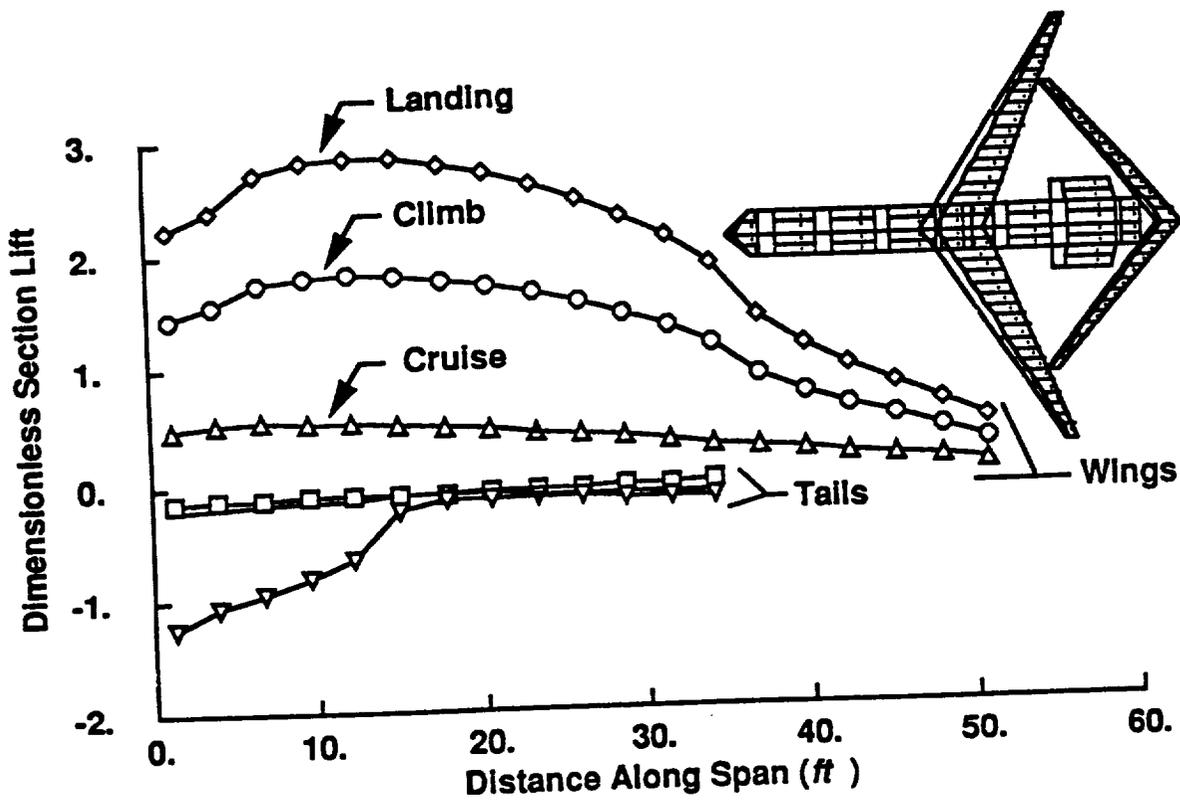
Fig. 1 Wing and tail lift distributions for a typical joined wing that is trimmed for climb, cruise, and landing approach.
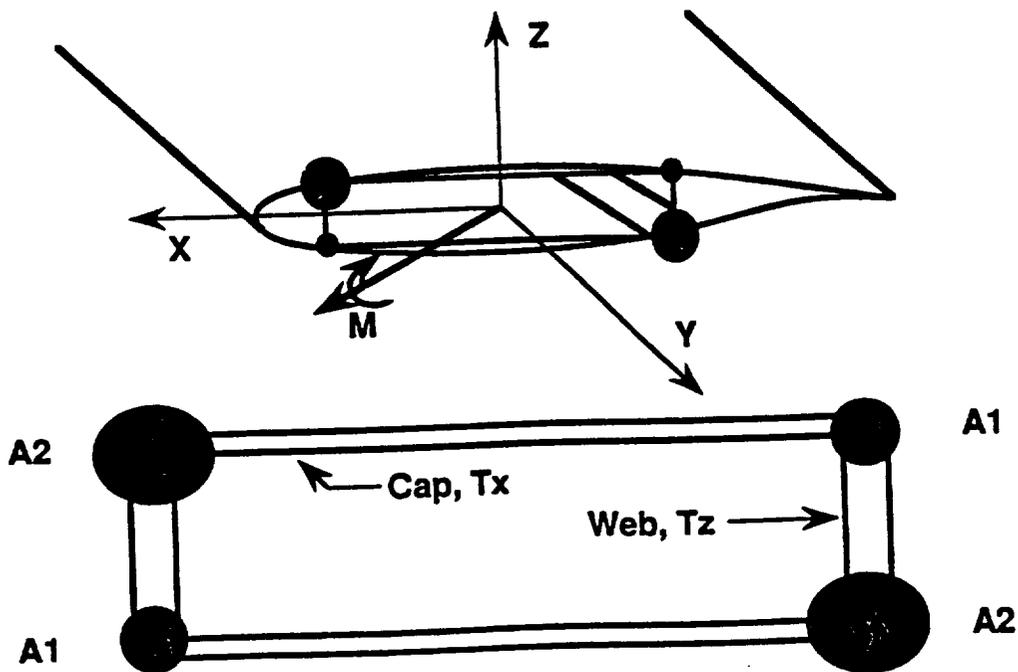
Fig. 2 Structural box model with skins, stringers, and the resultant bending moment.
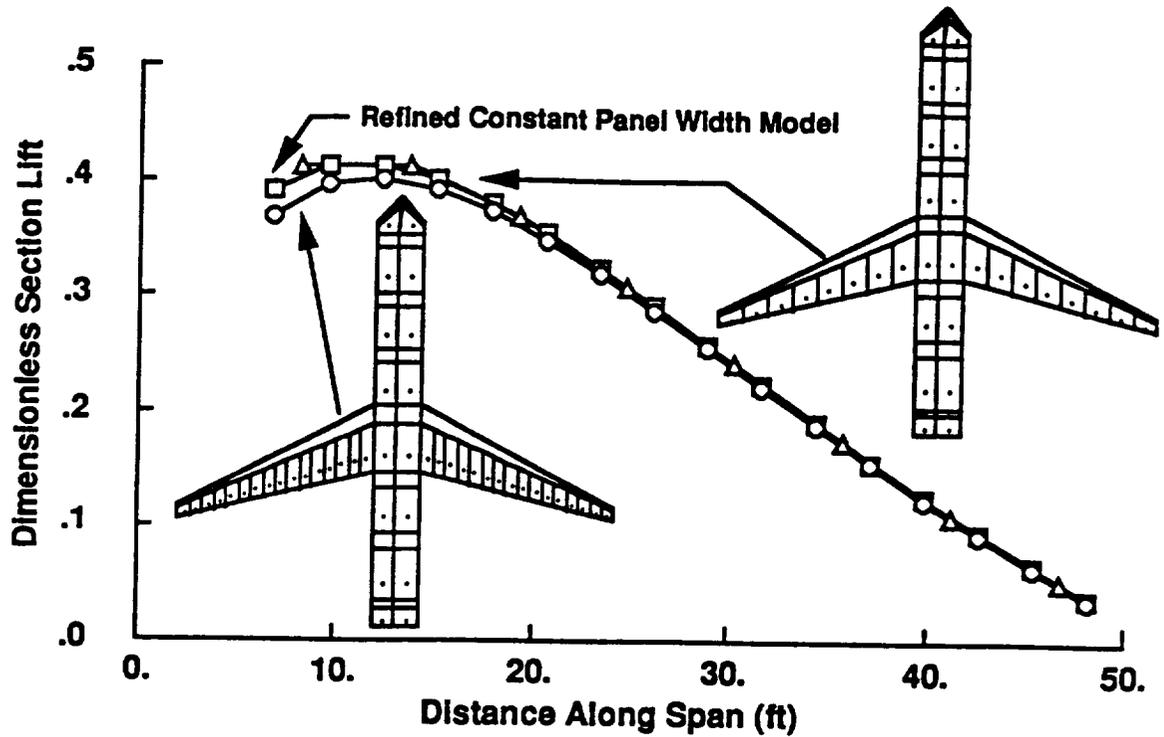
Fig. 3 Lift distributions calculated with and without a panel width discontinuity at the wing-fuselage joint.
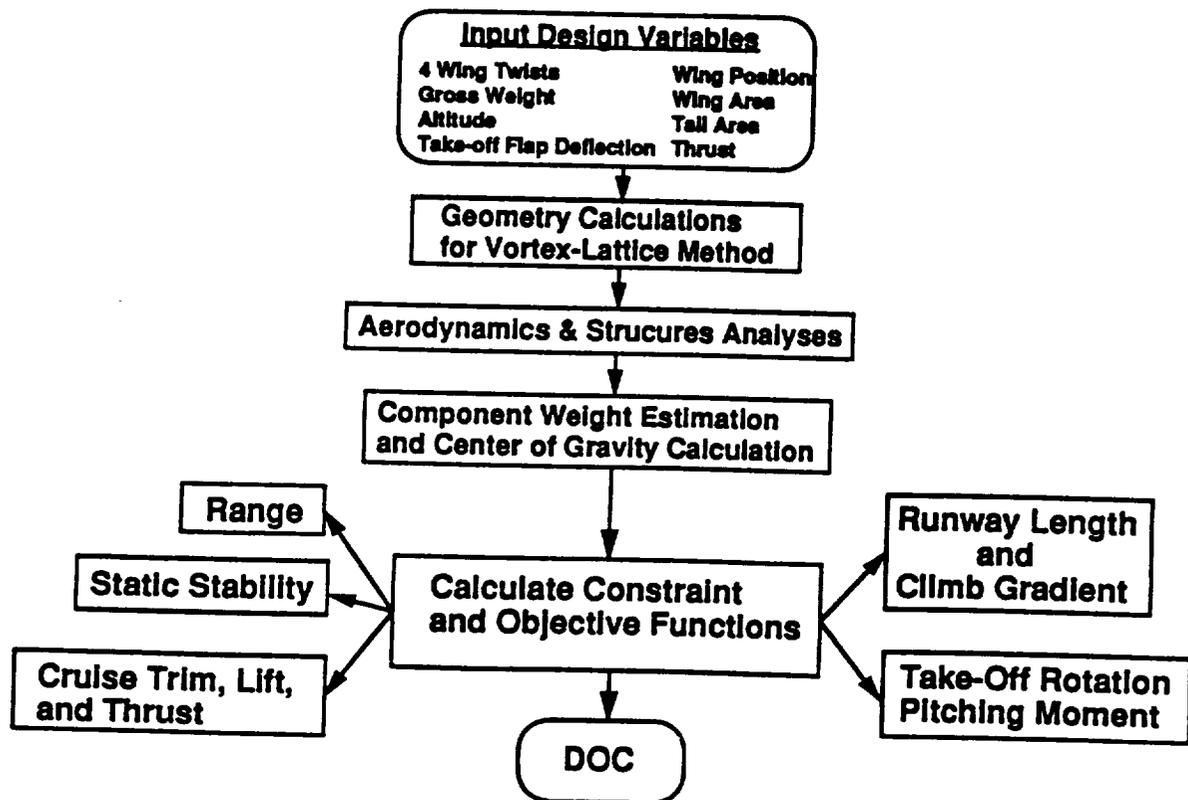
Fig. 4 Design variables, constraints, and a flow chart of the calculations in the design code.
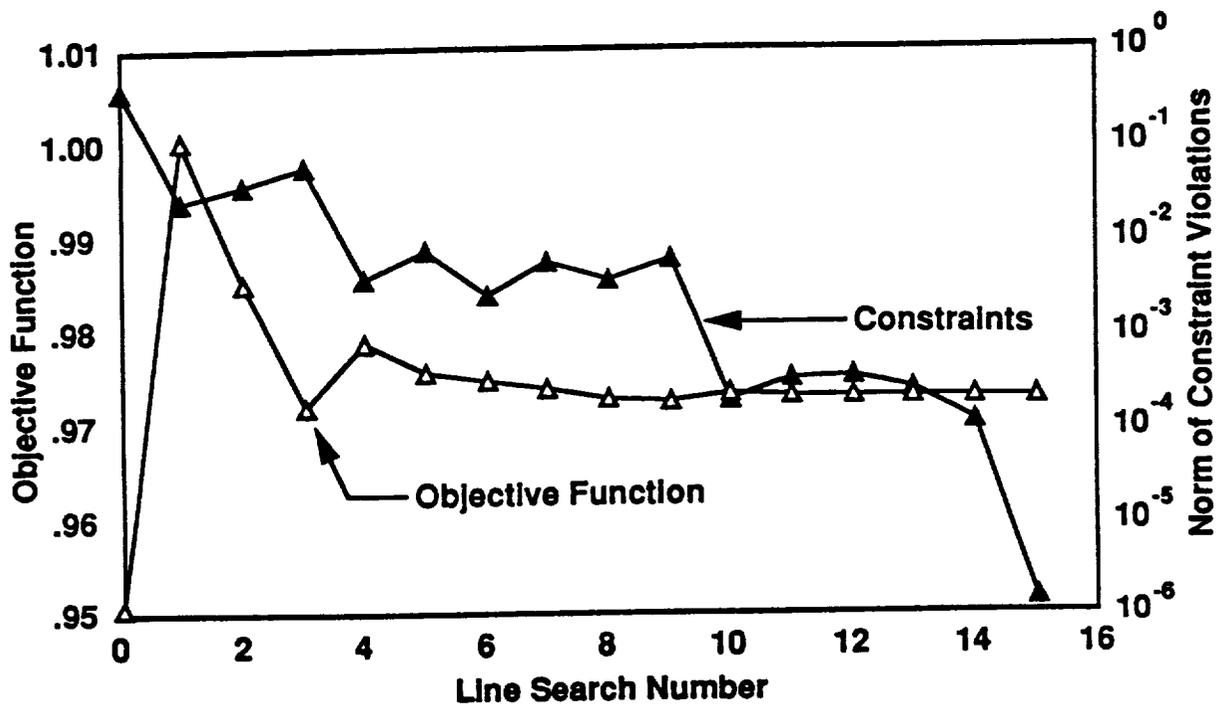
Fig. 5 The iteration history of the objective function and the constraint violations for a joined-wing design problem.
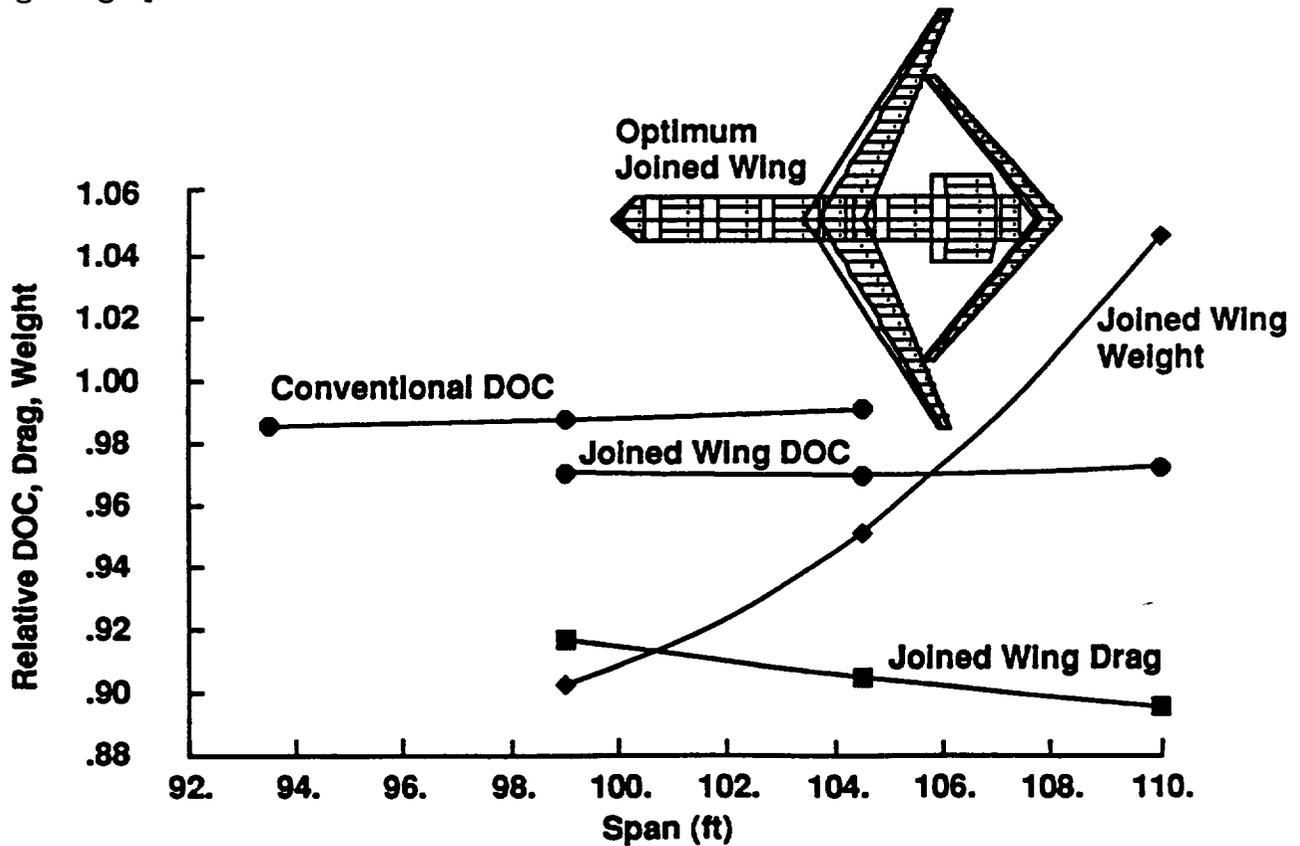
Fig. 6 Optimization results for conventional T-tail and joined-wing transports.

444

LARGE SCALE NONLINEAR NUMERICAL OPTIMAL CONTROL FOR FINITE ELEMENT MODELS OF FLEXIBLE STRUCTURES

Christine A. Shoemaker and Li-zhi Liao
Cornell University

Abstract

This paper discusses the development of large scale numerical optimal control algorithms for nonlinear systems and their application to finite element models of structures. This work is based on our expansion of the optimal control algorithm (DDP) in the following steps: a) improvement of convergence for initial policies in nonconvex regions; b) development of a numerically accurate penalty function method approach for constrained DDP problems; and c) parallel processing on supercomputers. The expanded constrained DDP algorithm was applied to the control of a four-bay, two dimensional truss with 12 soft members, which generates geometric nonlinearities. Using an explicit finite element model to describe the structural system requires 32 state variables and 10,000 time steps. Our numerical results indicate that for constrained or unconstrained structural problems with nonlinear dynamics, the results obtained by our expanded constrained DDP are significantly better than those obtained using linear-quadratic feedback control.

I. Introduction

The discrete-time optimal control problem studied in this paper has the following format:

$$\min_{(u_1, \cdots, u_N)} \sum_{t=1}^{N} g(x_t, u_t, t) \tag{1.1}$$

$$\text{where} \quad x_{t+1} = f(x_t, u_t, t) \quad t = 1, \cdots, N-1, \tag{1.2}$$
$$x_1 \equiv \bar{x}_1 \quad \text{given and fixed}$$

where $x_t \in R^n$ and $u_t \in R^m$ are state and control variables; the function $g : R^{n+m+1} \to R^1$ is the objective function (or performance index); and the function $f : R^{n+m+1} \to R^n$ is the transition function. In some cases, there are constraints on state and/or control variables of the form

$$h(x_t, u_t, t) \leq 0 \quad t = 1, \cdots, N, \quad \text{for } h : R^{n+m+1} \to R^p. \tag{1.3}$$

In the following analysis, all of the functions g, f and h may be nonlinear. The differential dynamic programming (DDP) algorithm first suggested by Mayne [6] and expanded by Jacobson and Mayne [1] addressed unconstrained problems of the form (1.1)–(1.2). Ohno [8] and Yakowitz [11] introduced methods for constrained differential dynamic programming (CDDP) problems. Yakowitz's theoretical development is limited to linear constraints and has been implemented on sizable problems with linear constraints (Jones et al [2]). Ohno's method has not been numerically implemented on large scale problems.

Liao and Shoemaker [5] recently proposed a new algorithm, which is called Constrained Differential Dynamic Programming (CDDP), for solving general nonlinear constrained problems. This algorithm is based on penalty function method, QR factorization and matrix partition techniques, and the unconstrained DDP algorithm. Numerical results of the CDDP algorithm

on some test problems with hundreds of state variables and nonlinear constraints on both state and control variables were also presented in [5]. Liao [3] has proved that 1) when the penalty parameter approaches its limit, the CDDP algorithm is globally convergent and converges to a local minimum; and 2) the CDDP algorithm does not suffer from the numerical problems associated with the traditional penalty function method. The leading terms in the computational complexity for the worst case of the CDDP algorithm per iteration are the following:

$$\frac{7}{2}n(n + m)^2 + \frac{16}{3}m^3 + p(n + m)^2 + \frac{1}{3}(m - s)^3 + \frac{1}{2}ms^2 + (m - s)^2s + 2(m - s)s^2 \qquad (1.4)$$

where s is an integer between 0 and m, and the forward simulation is not counted here (for details, see Liao [3]). Shoemaker $et\ al$ [10] have reported numerical results on parallel processing for the Liao and Shoemaker [5] constrained differential dynamic programming algorithm. Computational time was reduced by a factor of about two thirds with the use of six parallel processors on the Cornell Supercomputer, an IBM 3090-600E. Vetorization was very effective in reducing computation time.

II. A Nonlinear Structural Control Application

2.1 Description of a general structural control problem

There has been little literature on the optimal control of nonlinear structures, and to our knowledge, no previous research on the coupling of nonlinear optimization to nonlinear finite element structural models. There have been many applications of optimal control techniques to flexible structures but most of them use LQ (Linear system dynamics, Quadratic performance function) techniques that assume the system's dynamics are linear. The major advantage of LQ is that feedback control can be rapidly computed. Previous workers have used LQ control policies computed on the basis of a linear approximation of nonlinear dynamics hoping this control would work reasonably well on nonlinear systems.

The results presented here indicate that nonlinear DDP optimal control policies based on the nonlinear dynamics of a structure perform significantly better for nonlinear structures than do the conventional LQ control policies. Both the DDP and CDDP algorithms provide time-varying feedback control laws.

Following the description in Shoemaker $et\ al$ [9], the dynamics of the structure are described by:

$$M \cdot \ddot{d}_t + C \cdot \dot{d}_t + K(d_t) \cdot d_t = p_t + B(d_t) \cdot u_t \qquad (2.1)$$

where d_t is the vector of nodal displacements at time t, $d_t \in R^n$; $\dot{d}_t$ is the vector of nodal velocities at time t, $\dot{d}_t \in R^n$; $\ddot{d}_t$ is the vector of nodal accelerations at time t, $\ddot{d}_t \in R^n$; $K(d_t)$ is the stiffness matrix, $K(d_t) \in R^{n \times n}$; M is diagonal mass matrix, $M \in R^{n \times n}$; C is mass proportional viscous damping matrix, $C \in R^{n \times n}$; $B(d_t)$ is control influence matrix, $B(d_t) \in R^{n \times m}$; p_t is external load history vector, $p_t \in R^n$; and u_t is control forces vector, $u_t \in R^m$.

The criterion used to select the optimal policy is the discretization of the objective function suggested by Miller $et\ al$ [7]:

$$\sum_{t=1}^{N} \left(\theta_{K,t} d_t^T K d_t + \theta_{M,t} \dot{d}_t^T M \dot{d}_t + \theta_{B,t} u_t^T B^T K_0^{-1} B u_t \right) \qquad (2.2)$$

446

where the θ's are weighting parameters, $K_0 = K(0)$ and $d_t^T K d_t$, $d_t^T M \dot{d}_t$, $u_t^T B^T K_0^{-1} B u_t$ are measures of strain energy, kinetic energy and potential energy, respectively.

The terms K and B introduce geometric nonlinearities into the system because they are functions of d. In addition, it is necessary in some situations to put constraints on the control vector u_t of the form

$$u_t^{min} \leq u_t \leq u_t^{max}$$

(2.3)

where u_t^{min} and u_t^{max} are given parameters that reflect the range of force of the controller.

2.2 Formulation of the transition equation

Since the DDP and CDDP algorithms solve the discrete-time optimal control problem, we need to discretize the differential equation (2.1). After introducing difference equations in equation (2.1), we have

$$\begin{pmatrix} d_{t+1} \\ \dot{d}_{t+1} \end{pmatrix} = A_1 \begin{pmatrix} d_t \\ \dot{d}_t \end{pmatrix} + B_1 \begin{pmatrix} p_t \\ B \cdot u_t \end{pmatrix}$$

(2.4)

where

$$A_1 = \begin{pmatrix} I_n - h^2 M^{-1} K & h I_n - h^2 M^{-1} C \\ -h M^{-1} K & I_n - h M^{-1} C \end{pmatrix}, \quad B_1 = \begin{pmatrix} h^2 M^{-1} & h^2 M^{-1} \\ h M^{-1} & h M^{-1} \end{pmatrix},$$

(2.5)

and h is the time difference used in the difference equations.

The stiffness matrix K is a quadratic function of d_t, and can be expressed as

$$K = K_0 + \frac{1}{2} K_1 + \frac{1}{3} K_2$$

(2.6)

where K_0 is a constant matrix, K_1 is a matrix only having linear terms in d_t, K_2 is a matrix only having quadratic terms in d_t. Now, we define our state variable as

$$x_t = \begin{pmatrix} d_t \\ h \dot{d}_t \end{pmatrix} \in R^{2n} \quad \text{for} \quad t = 1, \cdots, N$$

(2.7)

and keep the control variable unchanged for the DDP and CDDP algorithms. Then, our final transition equation is

$$x_{t+1} = A_4 \cdot x_t + B_4 \begin{pmatrix} p_t \\ B \cdot u_t \end{pmatrix} \quad t = 1, \cdots, N-1$$

(2.8)

where A_4 and B_4 are derived from terms in (2.4) $-$ (2.6) as described in Liao [3]. The matrices A_4 and B are functions of x_t and hence (2.8) is nonlinear.

2.3 A four-bay nonlinear structural control problem

In this section, we will apply the theory discussed in previous sections to the optimal control of a four-bay, two dimensional truss with 12 soft members that generate geometric nonlinearities. There are two actuators that act parallel to the length of the structure; see Figure 1.

447
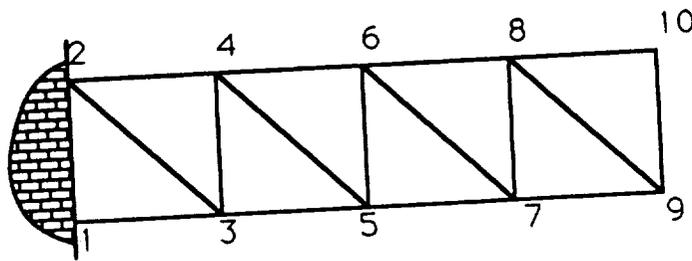
Fig. 1 A two dimensional four-bay truss.

Some Specifications of the Problem

1) The damping matrix C is assumed to be zero.

2) The control influence matrix B is constant.

3) The dimension of the nodal displacement vector $n = 16$, i.e. $d_t \in R^{16}$, then $x_t \in R^{32}$.

4) The dimension of the control force vector $m = 2$, i.e. $u_t \in R^2$.

5) The external load vector

$$p_t = (\underbrace{0, \cdots, 0, -0.07}_{14}, 0, 0)^T \in R^{16} \quad 1 \leq t \leq 757, \quad p_t = 0 \in R^{16} \quad t \geq 758. \tag{2.9}$$

6) The time interval in the explicit time-marching scheme $h = 0.0003299998$ sec.

7) The total time steps $N = 10,000$.

8) The weighting parameters $\theta_{K,t} = 1$, $\theta_{M,t} = 1$ and $\theta_{B,t} = 2$.

9) The units used in this truss are d_t (inch), $\dot{d}_t$ (inch/sec.), p_t (kip), u_t (kip), h (sec.), energy (inch·kip).

10) In the constrained case, the constraint function is $-0.05 \leq (u_t)_i \leq 0.18$, $i = 1, 2$, $t = 1, \cdots, N$.

The dynamics for the nonlinear system in Table 1 are generated by the dynamics of equation (2.8) with the actual stiffness matrix K, which is represented by equation (2.6). For the linearized system, however, the dynamics are simulated with equation (2.8) and the assumption that K is constant, i.e. $K = K_0$. The objective function (2.2) is the sum of strain, kinetic and potential energies. Our goal is to minimize this function which, since K is a second order function of x_t, results in a fourth order polynomial function of the state vector. The initial value of the DDP algorithm in Table 1 is the nonlinear system closed-loop linear feedback. The adaptive shift procedure (see Liao and Shoemaker [4]) was used. It took 44 iterations of the DDP algorithm to reach the optimal solution. Table 1 gives the values of objective function for the unconstrained system. We see that the total value of the objective function is reduced by 27% when the linearized feedback control is replaced by the optimal DDP solution although our experiments showed that the nonlinear system is not very nonlinear, i.e. uncontrolled trajectory does not differ very much between the nonlinear and linear systems.

Table 2 considers the case in which the control forces are constrained to be in the range from $u_t^{min} = -0.05$ to $u_t^{max} = 0.18$ kip. There is no linear theory for the constrained problem so the most direct way to use linear feedback is to apply the control force

$$u_t = Min(u_t^{max}, Max(u_t^{min}, \hat{A}_t + \hat{B}_t x_t)), \tag{2.10}$$

$$x_{t+1} = T(x_t, u_t, t)$$

<div align="right">(2.11)</div>

where $\hat{A}_t$ and $\hat{B}_t$ define the linear feedback control law for minimizing (2.2) given (2.7) assuming for both cases that K_1 and $K_2 = 0$ (i.e. $K = K_0$).

Table 1 Unconstrained Case: Table entries are values of the objective function (inch·kip) and their percentage contributions to the total.

	Strain	Kinetic	Potential	Total Energy
No Control Linear System	8639.54 (51.2%)	8222.65 (48.8%)	0.0 (0.%)	16862.2 (100.%)
No Control Nonlinear System	7964.70 (50.4%)	7832.03 (49.6%)	0.0 (0.%)	15796.7 (100.%)
Linear System with Linear Feedback	772.45 (27.5%)	494.93 (17.6%)	1536.82 (54.8%)	2804.19 (100.%)
NonLinear System with Closed-loop Linear Feedback	618.79 (26.2%)	364.86 (15.5%)	1376.30 (58.3%)	2359.94 (100.%)
Optimal Solution from DDP	668.93 (38.6%)	379.03 (21.9%)	684.61 (39.5%)	1732.57 (100.%)

Table 2 Constrained Case: Table entries are values of the objective function (inch·kip) and their percentage contributions to the total.

	Strain	Kinetic	Potential	Total Energy
Truncated Closed-loop Linear Feedback	989.54 (35.7%)	797.19 (28.8%)	984.46 (35.5%)	2771.19 (100.%)
Truncated Unconstrained Optimal DDP Solution	863.81 (41.3%)	634.04 (30.3%)	593.36 (28.4%)	2091.20 (100.%)
Optimal Solution from Constrained DDP	722.28 (38.6%)	432.26 (21.9%)	639.68 (39.5%)	1794.22 (100.%)

Equations (2.10) and (2.11) define the **truncated closed-loop linear feedback** policy in Table 2. The **truncated optimal DDP** policy is $u_t = Min(u_t^{max}, Max(u_t^{min}, u_t^*(x))$ where $u_t^*(x)$ is the optimal DDP solution to the corresponding unconstrained problem. The results of these policies are compared to the solution obtained using the constrained version of DDP. The initial value of the CDDP algorithm in Table 2 is the truncated unconstrained optimal DDP solution. The adaptive shift procedure (see Liao and Shoemaker [4]) was used. It took 45 iterations of the CDDP algorithm to reach the optimal solution. The results in Table 2 indicate that the policy computed by the constrained DDP algorithm is much better than the truncated linear policy since the objective function value is 35% lower for the optimal constrained DDP algorithm than for the truncated linear policy.

These results support the use of the DDP algorithm, especially in its constrained version, for active control of flexible structures since obviously better control is obtained. The advantages

of using nonlinear optimization techniques as opposed to a linear feedback would be expected to be even larger for a more nonlinear structure. The advantages of our algorithm are significantly enhanced when there are constraints on the controls since linear feedback theory assumes the controls are unconstrained.

III. Conclusions

Our numerical results indicate that substantial improvement in performance can be gained by using a nonlinear optimal control algorithm rather than by applying LQR theory to a linearized approximation of the nonlinear system. However, the computational requirements will be significantly larger for DDP than for LQR and hence increases in performance will require increases in computational resources.

Acknowledgements

References

[1] D. Jacobson and D. Mayne, *Differential Dynamic Programming*, Elsevier Sci. Publ., New York, 1970.

[2] L. Jones, R. Willis and W. Yeh, "Optimal control of nonlinear groundwater hydraulics using differential dynamic programming," *Water Resource Res.*, vol. 23, pp. 2097-2106, 1987.

[3] L-Z. Liao, "Numerically efficient algorithms for unconstrained and constrained differential dynamic programming in discrete-time, nonlinear systems," Ph.D. thesis, Cornell University, 1990.

[4] L-Z. Liao and C. A. Shoemaker, " Convergence in unconstrained discrete-time differential dynamic programming," *IEEE Trans. Automat. Contr.*, (accepted).

[5] L-Z. Liao and C. A. Shoemaker, " Partitioning and QR factorization to avoid numerical ill-conditioning of penalty functions for constrained nonlinear programming and control," (submitted)

[6] D. Mayne, "A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems," *Intnl. J. Control*, vol. 3, pp. 85-95, 1966.

[7] D. F. Miller, V. B. Venkeyya and V. A. Tischler, "Integration of structures and controls - some computational issues," Proceedings of 24th Conf. on Decision and Control, pp. 934-941, 1985.

[8] K. Ohno, "A new approach of differential dynamic programming for discrete time systems," *IEEE Trans. Automat. Contr.*, vol. AC-23, pp. 37-47, 1978.

[9] C. A. Shoemaker, L-Z. Liao, B. Aubert and J. F. Abel, "Optimal nonlinear control with geometrically nonlinear finite element analysis for flexible structures" in *Computational Mechanics*, A. N. Atluri and G. Yagawa (editors), Springer-Verlag, 1988.

[10] C. A. Shoemaker, L-Z. Liao, H. Caffey and L-C. Chang, "Optimal control of large scale nonlinear engineering systems," IBM Supercomputing Competition, 1990 (in press).

[11] S. Yakowitz, "The Stagewise Kuhn-Tucker Condition and Differential Dynamic Programming," *IEEE Trans. Automat. Contr.*, vol. AC-31, pp. 25-30, 1986.

PAYCOS: A NEW MULTIDISCIPLINARY ANALYSIS PROGRAM FOR HYPERSONIC VEHICLE DESIGN

J. R. STUBBE
Lockheed Missiles and Space Company
Sunnyvale, California

S64-05

P. 6

ABSTRACT

The Payload Conceptual Sizing Code (PAYCOS), a new multidisciplinary computer program for use in the conceptual development phase of hypersonic lifting vehicles (HVs), is described. The program, written at Lockheed Missiles and Space Company in Sunnyvale, California, allows engineers to rapidly determine the feasibility of an HV concept and then improve upon the concept by means of optimization theory.

The code contains analysis modules for aerodynamics, thermodynamics, mass properties, flight stability, controls, loads, structures, and packaging. Motivation for the code lies with the increased complexity of HVs over their body-of-revolution ballistic predecessors. With these new shapes, the need to rapidly screen out poor concepts and actively develop new and better concepts is an even more crucial part of the early design process.

Preliminary results are given which demonstrate the optimization capabilities of the code.

INTRODUCTION

The detailed design of hypersonic lifting vehicles (HVs), such as a Maneuvering Reentry Body (MaRB), is a complex and time consuming task, involving many people and years of effort. Such a design effort consists of concept development, preliminary design, detailed design and finally the fabrication of a prototype or flight test vehicle.

In the early design phases, beginning with concept development and continuing into preliminary design, the emphasis is on evaluating the capabilities of many competing designs. To accomplish this requires a synthesis of aerodynamics, thermodynamics, mass properties, flight stability, controls, loads, structures, and packaging (i.e. a multidisciplinary approach).

Recently, effort has been directed into developing multidisciplinary analysis programs for use during the preliminary design phase of aerospace structures[1,2]. When coupled with optimization theory, these tools are a powerful means of obtaining better designs. Unfortunately, preliminary design tools are generally not well suited for concept development. This is due in part to the differences between concept development and preliminary design.

At the concept development phase, the emphasis is on obtaining an HV external shape that satisfies various performance requirements. The vehicle can't be too long, too short, too heavy, too stable, too unstable, etc. Each of the flight disciplines has to be considered in a global manner (that is, how one discipline effects, either positively or negatively, the others). Once suitable shapes have been found, the preliminary design phase begins.

In preliminary design, it is structural optimization which becomes important. The analysis becomes more focused; constraints are tightened, subsystems are defined and analyzed, and complex phenomena such as buckling and flutter are studied, all of which lead to a general increase in pre-processing, run, and post-processing time. Thus, preliminary design demands a greater degree of detail than is required for concept development and takes far longer than a small analysis team has in the short amount of time allocated for HV concept development.

The purpose of the Payload Conceptual Sizing Code (PAYCOS) is to bring multidisciplinary analysis and optimization directly to the concept development level and give engineers the ability to screen out poor designs and actively pursue promising designs prior to the preliminary design effort.

OVERVIEW

From the beginning, it was realized that PAYCOS had to be: a) rapid, b) modular, c) accurate, and d) portable. PAYCOS had to rapidly perform the data synthesis, and hence the sizing process, to allow a great many configurations to be analyzed in a short amount of time. PAYCOS had to be modular so that the code can be easily modified and enhanced. PAYCOS had to have sufficient accuracy so that the zero and first order concept level results would carry through to detail design. Finally, PAYCOS had to be able to run on a wide variety of computer platforms with little or no changes to the code.

PAYCOS is capable of sizing a wide range of hypersonic vehicles from body-of-revolution derivatives (Figure 1a) to complex glide vehicles (Figure 1b).

(a) Body-of-Revolution Derivative   (b) Glide Vehicle
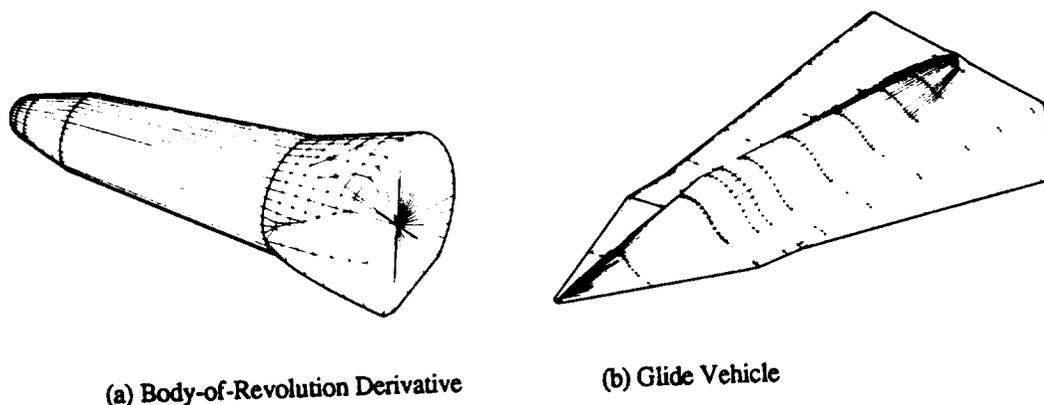
Figure 1: Common Hypersonic Lifting Vehicles

The code consists of modules for each of the major flight disciplines. A brief description of each Module is as follows:

Startup Module - To provide the estimates required to "kickstart" the analysis modules.
Geometry Generation Module - To provide a geometric exterior shape through a discrete number of input variables.
Panel Generation Module - To mesh the external geometry with flat four noded panels.
Aerodynamic Module - To calculate aerodynamic coefficients for each geometric panel.
Static Survey Module - To determine the flight stability of the vehicle.
Controls Module - To determine control system requirements.
Mass Properties Module - To provide mass and mass moment of inertia distributions.
Packaging Module - To provide payload and subsystem packaging requirements.
Loads Module - To provide external pressures and axial, shear and moment distributions.
Structures Module - To provide skin and internal structure weight and volume.
Optimization Module - To improve the original design.
Post Processing Module - To provide sorting, plotting and comparison capabilities.

The PAYCOS modules are based as much as possible on known analysis techniques and tools. As an example, the Aerodynamics Module uses SIMP, a specially modified version of SHAB[3] (the Supersonic-Hypersonic Arbitrary Body Program, developed at McDonnell-Douglas). Whenever new coding is required, as was the case for the Mass Properties, Packaging and Structures Modules, a thorough technical validation is performed.

The coding of each module follows a set of common software standards. The modules are connected through software framework that is monolithic (see Figure 2) for fast execution time, yet supports a command language and database management system compatible with existing software at Lockheed Missiles and Space Company[4]. All computer coding is done in ANSI standard Fortran 77.
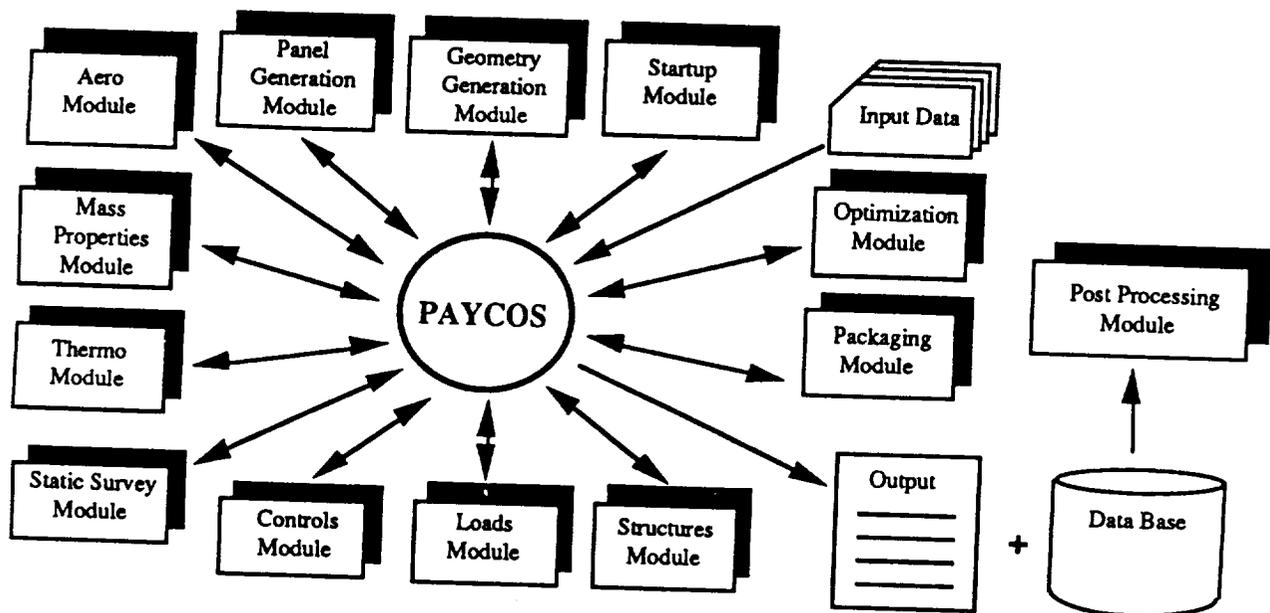
Figure 2: The PAYCOS Hypersonic Vehicle Analysis Program

To date PAYCOS has been successfully run on VAX, IRIS and CONVEX computers. Execution time is well within acceptable limits for rapid concept evaluation. A single analysis takes approximately sixty seconds on a VAX 780, twenty seconds on a Personal IRIS Workstation, and four seconds on a CONVEX C220. Very few problems were encountered during the porting process.

Optimization Capabilities

PAYCOS is used in two capacities: a) to rapidly analyze a given concept, and b) to improve upon that concept through optimization theory. The heart of the Optimization Module is ADS[5], the Automated Design Synthesis program which has been used successfully throughout the Aerospace Industry. Commands are available which give the user complete control over the optimization process including save, continue and restart capabilities. Additional capabilities, like looping and conditional checks, are made available through a Fortran-like language[6] that can be executed from inside the data deck.

453

The optimization problem solved in PAYCOS is expressed as

Minimize $\quad f(X)$

Subject to:

$$g_j(X) \leq 0; j = 1,m$$

$$X^l_i \leq X_i \leq X^u_i; i = 1,n$$

Where **X** is a set of bounded input variables (design variables), f(X) is a single function (objective function) being minimized (or maximized), and $g_j(X)$ are the inequality constraints.

PAYCOS allows for a good deal of flexibility when it comes to the optimization problem statement. The Optimization Module contains a library of candidate objective and constraint functions. A partial list of available objective and constraint functions is given in Table 1.

Table 1	
Available constraint/objective functions	
1 Total weight	6 Roll stability
2 Lift/drag	7 Internal volume
3 Yaw stability	8 CG = Cp + offset
4 Roll efficiency	9 Flap loads
5 Ballistic coefficient	10 Trim

Design variable definition is completely up to the user. Typically, the PAYCOS design variables are the inputs to the Geometry Generation Module and the optimization process is one of aeroshaping. The Geometry Generation Module consists of a number of design specific geometry generators. This means that a body-of-revolution derivative has a different geometry generator than a delta wing "glider", which has a different geometry generator than a dart shaped "evader". This is in lieu of much more difficult task of developing a single universal generator that lends itself to efficient aeroshape optimization. A single flag is used to tell the Geometry Generation Module which generator to use.

Preliminary Optimization Results

The Optimization Module containing ADS has been successfully linked with PAYCOS and a variety of optimization problems have been successfully run. In Figure 3, the iteration histories of two functions are given for one such test case. Quanitity values have been intentionally deleted. In Figure 3a, yaw stability (the objective function) is shown. In Figure 3b, lift/drag (one of the constraints) is shown. Note that although the test vehicle was made more yaw stable, as was the goal, it was done so at the cost of lift/drag.
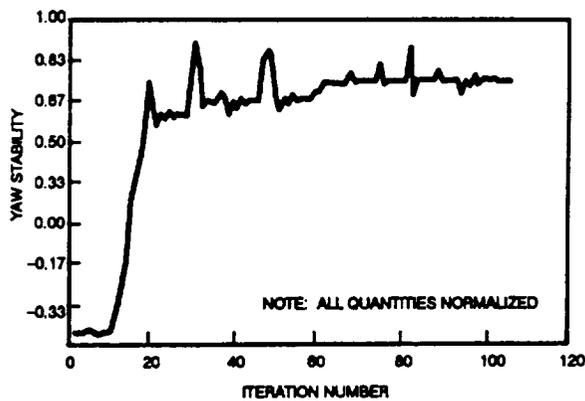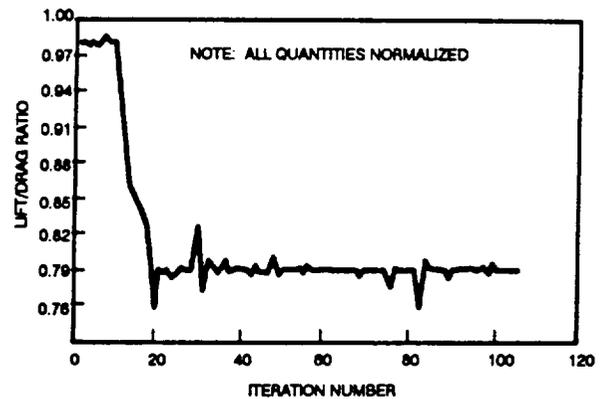
Figure 3: Iteration History for Maximizing Yaw Stability

Of course, obtaining better designs is not the only value of using optimization. In concept development, a "best" design can be considered secondary to understanding how a change in one function affects another. The large amount of data generated during the optimization run can be used to generate the desired sensitivity plots. Consider Figure 4a, where ballistic coefficient is plotted against yaw stability, and Figure 4b, where roll efficiency is plotted versus vehicle weight for the optimization problem of Figure 3. The trends are obvious: As yaw stability increases, ballistic coefficient decreases, and as weight increases, roll efficiency remains fairly flat.
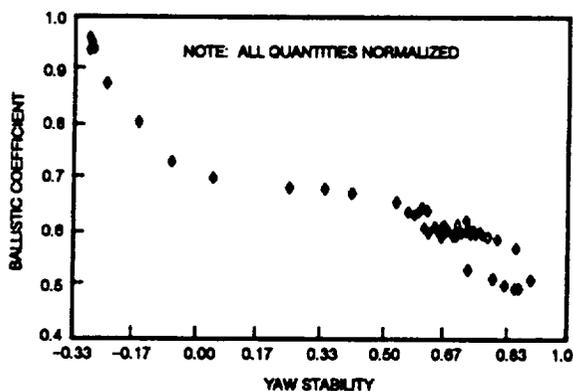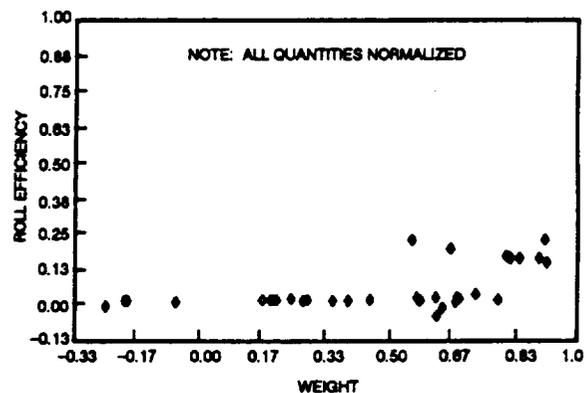
Figure 4: Sensitivity Comparison Between Functions

Part of the validation process involves tailoring ADS to the types of optimization problems PAYCOS is used to solve. From the start, the preferred optimization method was the modified method of feasible directions[7] with polynomial interpolation used for the one dimensional search (ISTRAT= 0, IOPT= 5, INOED= 7) because of its robustness in handling initially infeasible designs. The relative convergence criteria has been relaxed to a 5% change in objective function (WK(12) = .05) with the absolute convergence criteria (WK(8)) set according to the chosen

455

objective function. To avoid having to create a Sensitivity Module, the internal ADS finite difference routine is used to calculate the function gradients. A finite difference step size of 5% (WK(21) = .05) is used in most cases.

Conclusion

PAYCOS brings optimization directly to the concept level in a multidisciplinary hypersonic vehicle analysis code designed from the start to be used very early on in the design process. PAYCOS greatly increases the ability to rapidly generate feasible and improved hypersonic lifting body designs by providing not only designs themselves but also the means of understanding sensitivity of competing design requirements.

Version 1.0 of PAYCOS is scheduled for completion in Fall 1990.

References

1. Neill, D.J., Johnson, E.H., and Canfield, R., "ASTROS - A Multidisciplinary Automated Structural Design Tool," AIAA paper no. 87-0713, AIAA 28th Structures, Structural Dynamics and Materials Conference, Part 1, Monterey, California, April 6-8, 1987.

2. Abdi, F.F., Austel, L.G., and Chang,P.W., "An Approach to an Aero/Thermal/Elastic Design System," AIAA paper no. 88-2383, AIAA 29th Structures, Structural Dynamics and Material Conference, Williamsburg, Virginia, April 18-20, 1988.

3. Gentry, A.E., Smyth, D.N., and Oliver, W.R., "The Mark IV Supersonic-Hypersonic Arbitrary Body Program," AFFDL-TR-73-159 Volume 1, November 1973.

4. Wong, D.G. and Fuller,C.R., "Integrated Structural Analysis For Rapid Design Support," AIAA paper no. 86-1010-CP, 27th Structures, Structural Dynamics and Materials Conference, San Antonio, Texas, May 17-21, 1986.

5. Vanderplatts, G.N., Sugimoto, H., and Sprague, C.M., "ADS-1: A New General-Purpose Optimization Program," AIAA Journal, Vol. 22, No. 10, October 1984.

6. Felippa, C.A., "A Command Language for Applied Mechanics Processors," Lockheed Missiles and Space Company, Sunnyvale, California, LMSC Report no. D878511, 1981 (rev. 1985).

7. Vanderplatts, G.N., "An Efficient Feasible Direction Algorithm for Design Synthesis," AIAA Journal, Vol. 22, No. 10, October 1984.

An Information Driven Strategy to Support Multidisciplinary Design

Ravi M. Rangan*, *Graduate Research Assistant*
Robert E. Fulton, *Professor*

N94- 71480

George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332

Introduction

The design of complex engineering systems such as aircraft, automobiles and computers is primarily a cooperative multidisciplinary design process involving interactions between several design agents. The common thread underlying this multidisciplinary design activity is the information exchange between the various groups and disciplines. The integrating component in such environments is the common data and the dependencies that exist between such data.

This may be contrasted to classical multidisciplinary analyses problems where there is coupling between distinct design parameters. For example, they may be expressed as mathematically coupled relationships between aerodynamic and structural interactions in aircraft structures; between thermal and structural interactions in nuclear plants, and between control considerations and structural interactions in flexible robots. These relationships provide analytical based frameworks leading to optimization problem formulations.

However, in multidisciplinary design problems, information based interactions become more critical. Many times, the relationships between different design parameters are not amenable to analytical characterization. Under such circumstances, information based interactions will provide the best integration paradigm. i.e. : there is a need to model the data entities and their dependencies between design parameters originating from different design agents. The modeling of such data interactions and dependencies forms the basis for integrating the various design agents.

Disparate CAD/CAM systems and Data Management Problems

There is a large body of information and expertise that must be harnessed and made available both during, and subsequent to the design stages. The information that is to be made available to aid a particular designer is very difficult to handle - it is widespread, diffuse and unorganized [ALEX64], and originates from different sources and is conveyed in different mediums. This information is usually not accessible in a manner that permits the designers to best utilize it.

Progress in the development of computer aided design and computer aided manufacturing (CAD/CAM) systems has largely been driven by specific task automation needs. Drafting, conceptual solid modeling , finite element analysis, tolerance analysis, simulation of motion, tool path simulation, numeric control, automated process planning systems are among several tasks that have been significantly computerized. In such CAD/CAM environments, the designer's principal directive is to generate a geometric model of the product. This model serves as the primary input to several of these analyses, simulation and validation systems. Current CAD/CAM environments are largely driven by this master geometry model. *However, the issue of allowing these CAD systems to interact with data generated by the engineering design process has not been investigated from the standpoint of providing collaborative capabilities within these CAD systems.* This paper will attempt to study this interaction in more detail. Within this context, we shall introduce the notion of the design action:

---

* currently with *Structural Dynamics Research Corporation*, Milford, OH- 45151

the design action is the designer's point of influence on the product. It is introduced here as a means of referring to the interaction between the designer and the computer tool (in this case the CAD system) as it relates to the prevailing design process.

In order to externalize the designer's information burden in such environments, our objectives are :

- To study the engineering design process from the standpoint of information flows between interacting design agents

- Define a framework to integrate the design process with the design action - specifically:
  - Provide conceptual centralization of design life cycle data
  - Define a design environment to facilitate interaction with related design agents
  - Ensure data consistency between the design environment and the centralized life cycle databases

Scope of this Initiative

We shall investigate these issues from the standpoint of the CAD system as the design action. A similar set of capabilities may be applied to a design action such as an FEM program. The key paradigm is to integrate the design action with the prevailing design process and to externalize the information management capabilities relative to the multidisciplinary data interactions. The following methodology was used to address these issues:

- Identification of a thin slice of the problem domain for detailed study
- Data Collection using protocol studies
- Data Analysis and data modeling
- Prototype Development research issues
- Prototype Implementation

The design and manufacturing activities at a major computer manufacturing plant provided the thin slice for the case study. The multidisciplinary design considerations were investigated by studying the interactions between design groups such as mechanical design (sheet metal housing design), hardware board design (VLSI boards) and components engineering (various component specifications such as disk drives, switches, fans, line filters, power supplies, etc.). After identifying the significant data entities and their dependencies, entity relationship models [TEOR86] and constraint networks [SRIR89] were defined to capture the interdependencies. The entity relationship model identifies *cardinality* and *entity-attribute* relationships between data entities or objects. The constraint network models the various compatibility relationships between these data entities. These relationships may be formula based or heuristic based. These models are adapted to the geometric objects represented in CAD systems by allowing textual attributes to be attached to the geometric objects. This enhanced representation permits linkages to be established between different geometric entities. This is the basis for the collaborative interaction between the CAD system design action and the data entities generated by the prevailing design process.

The Design Model: Data and Constraints

Design organizations consolidate several active design agents that may work in a concurrent or serial mode and may or may not be independent. Pahl and Beitz [PAHL84] stress that the need for interdisciplinary collaboration is imperative to design in large scale organizations. Further, they note that the ready availability of a wide range of comprehensive and problem-oriented information is of utmost importance in the design process. The integration problem is to develop strategies to support the collaboration of design agents in solving complex

tasks. Here we assume that design agents are assigned to subtasks decomposed along discipline and functional boundaries and may require different problem solving skills, and methodologies. The purpose of this collaboration is to reduce the uncertainty and the resulting contingencies associated with the subtasks.

Let us consider a design agent, A. The input data is represented by Δ_i and is made available to A by the various upstream functions, U. The design agent acts on this information and transforms this data based on the operator characteristics (Θ_i) into output data (Δ_o) that is communicated to the downstream functions, D. In order to ensure that the operator's (Θ_i) characteristics are consistent with both the upstream and the downstream functions, it is necessary to impose a constraint function Ψ_i that incorporates the data interdependencies that exist between the upstream functions, U, the current function, A, and the downstream functions, D. Figure 1 shows a simple model describing this requirement. Δ_i, Δ_o are the data inputs and outputs with respect to Θ_i.
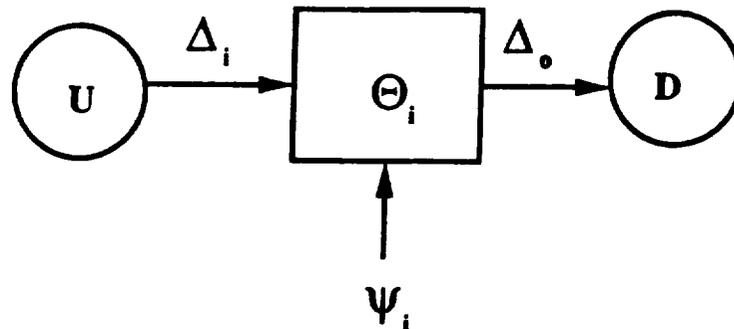
Figure 1 : Design - Data Input / Output Model

In the present case, the operator (Θ_i), represents the designer and the design action. The design action must therefore be capable of sharing data with the serial/ parallel functions. In order to provide such externalization capabilities at the design action, the design action must allow the designer to interact with the design process in an effortless manner. Figure 2 illustrates the nature of such an environment.
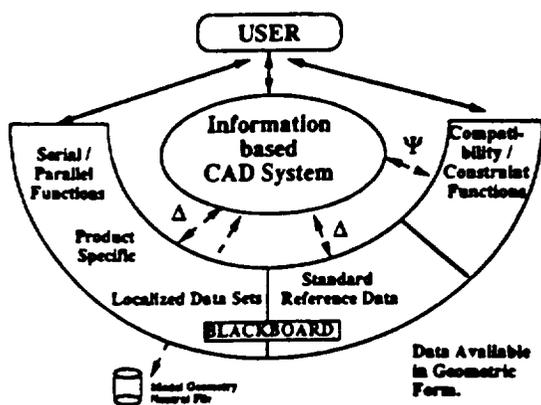
Figure 2 : Integration of Design Action
with Environment
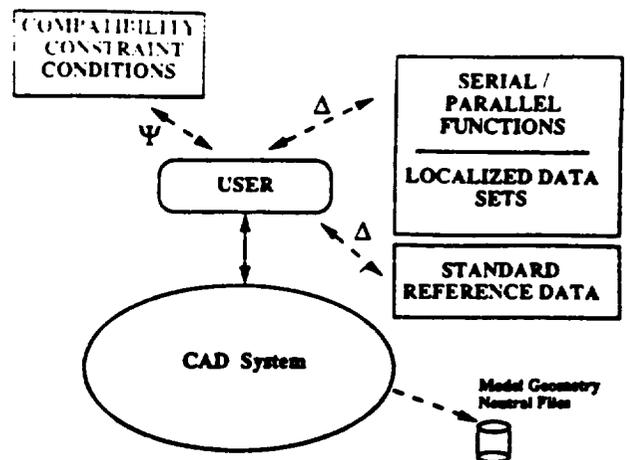
Figure 3 : Manual Interaction Between Designer
and CAD/CAM Environment

The user interacts with a CAD system, which in turn is intimately integrated with both the geometric database as

459

well as a blackboard. The life cycle data is maintained by several interacting disciplines and is indicative of the current state of the design and may be viewed as an information blackboard. Such a configuration ensures that the design action has access to up-to-date design data originating from different participants or teams in a distributed design environment. It must also be possible to manipulate these databases from the design action.

The compatibility requirement between the life cycle data and the geometric database dictates that changes in one system trigger changes in the other system as appropriate. This configuration significantly externalizes the data communication and consistency requirement that is required of design processes involved in multidisciplinary design by teams. This contrasts to the manual interaction mode illustrated in figure 3 which introduces inefficiency in the design process.

Proposed Architecture

Given the scope of this paper, this section describes an architecture to provide such externalizing capabilities at the CAD system (the design action).
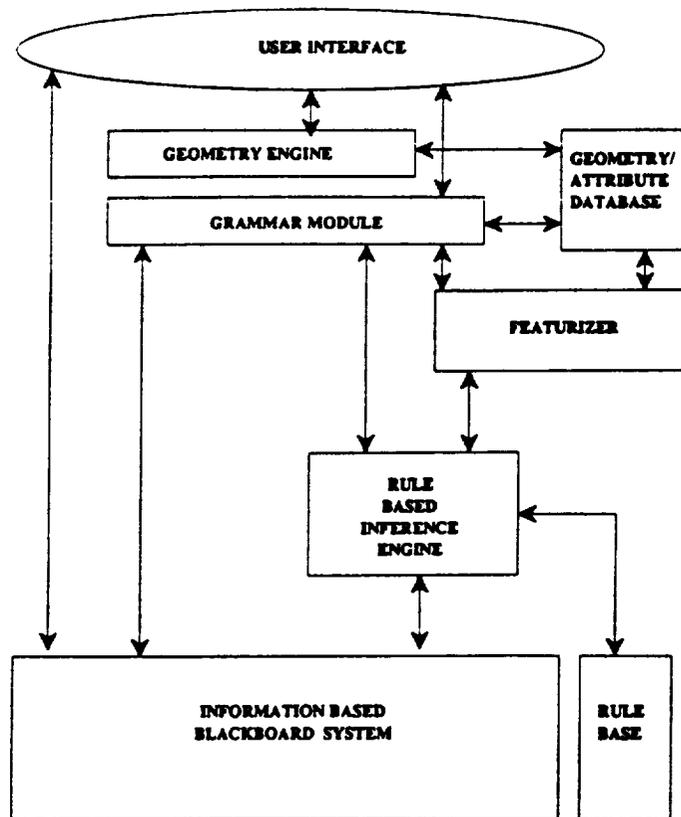
Figure 4 : Integrated Framework

The integrated framework supports an inference engine, a grammar module, a geometry database, a blackboard DBMS, a featurizer and a geometry engine. The geometry engine manipulates the geometry database (only geometry) directly, whereas the featurizer manipulates the geometry database as well as the attributes as per the grammar modules' or inference engines directives. The grammar module retrieves data from the geometry / attribute database as well as the blackboard database and manipulates the blackboard data and the geometry/attribute directly through the featurizer. The inference engine defines rule activations using facts derived from the geometry/attribute database and/or the blackboard data and may change the blackboard data directly and may also manipulate the geometry/attribute database via the featurizer. The components of the integrated framework are listed below:

460

Geometry / Attribute Database: The geometric entity is an aggregation of attribute values and a geometric object. In figure 5, the aggregation refers to the collection of primitive geometric elements which are combined in a specific manner to define the geometric entity. The description is a collection of attributes that describe the geometry. Furthermore, relationships between the attributes and the geometric object must be made available as appropriate. The entity is identified by pointer and type identifications. The attributes may provide information about positional and dimensional aspects (location, length) or about the linkages between different geometric entities (adjacency, connections) as well as descriptive data (voltage, power, part#).
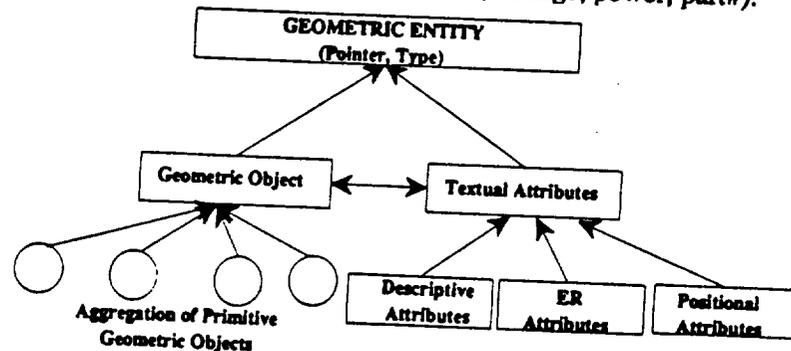
Figure 5: Geometry + Description Schema

Information Based Blackboard System: A blackboard database is essentially a shared database that defines the state of the product design to an expandable community of design agents. This state is an aggregation of the contributions from several design agents. The blackboard therefore provides a forum for tracking the status of the design at any point in time. In addition the blackboard provides control strategies to ensure consistency for data originating from several design agents. This conceptually centralized database stores data from previous designs, change requests, notes and related product information. Figure 6 illustrates the blackboard model.
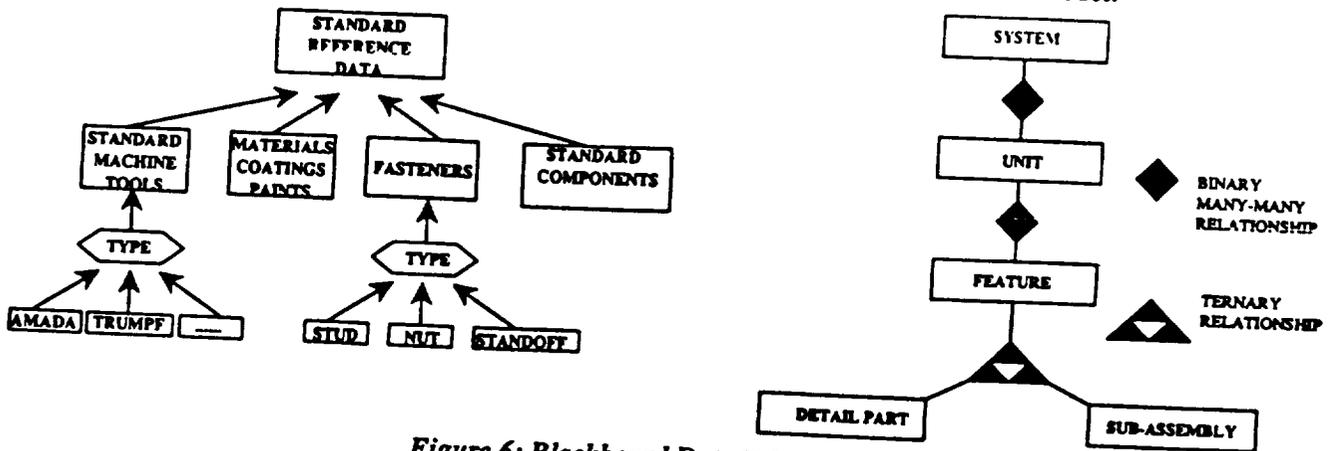
Figure 6: Blackboard Data Schema

In the current implementation a relational DBMS is used. The representation follows the entity relationship model and supports generalization and subset hierarchies (standard parts, tools and components) and a dynamic BOM hierarchy. The BOM structure permits data from different design agents to be represented in a manner consistent with the needs of the current design action. This has been implemented using the ORACLE DBMS [ORAC87] and is explained in [RANG89].

Grammar Module: The grammar module ensures consistency between the attributes defined in the geometric entity and the data in the blackboard structure. It also ensures consistency between the relationships defined in the ER attributes and facilitates automatic updates and constraint propagation effects relative to data defined in the standards databases and the attribute geometry database.

461

The grammar module is so called because it uses the shape grammar formalism [STIN80] to manipulate the geometry and the attributes. For example, in Figure 7, the square object with attributes A, B is replaced with three other objects with attributes : C, D, E, F, G. This transformation is brought about by rules which will depend on the square, A, B and the current state (defined by the contents of the centralized database) of the design process. Thus, the grammar module interacts with both the databases and provides a uniform mechanism for ensuring consistency between the design action and the blackboard structure.
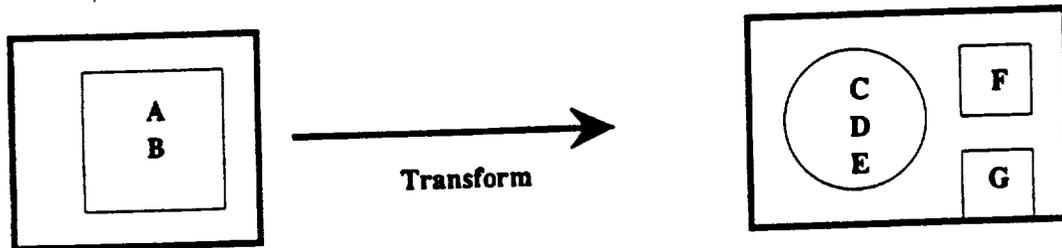
Figure 7: Example Grammar Operation

Featurizer: The featurizer transforms parametrized blackboard data (e.g. attributes of standard components) into geometric entities. In other words, it converts the geometric / attribute data into equivalent blackboard structures and vice versa.

Rule based Inference Engine: The CLIPS expert system shell was used to implement the rule base. This system is coded in C and was interfaced to the CAD system and the DBMS. A detailed account of the CLIPS implementation is discussed in [RANG190].

Geometry Engine / User Interface: In the prototype implementation, the AUTOCAD [AUTO89] system was used to simulate the CAD system. The customization features in Autocad were used extensively to augment the geometry based capabilities.

Capabilities of Implemented Prototype

The software prototype implementation is discussed in [RANG90]. Some of the key functionalities of the CAD system are enumerated below

- *Attachment of textual attributes to geometric entities: Encapsulation of form and function.*
- *Query facility for geometric entities : e.g.: select all holes > 0.5"*
- *Blackboard interaction with CAD system:*
  Automated information transfer between serial and parallel groups :
    insert geometric entity from standards DBMS
    insert geometrical entity from product specific Data
    (automated UPDATES, DELETIONS, INSERTS of dependent data entities)
- *Direct query link between CAD system geometric definition and DBMS attributes*
- *Maintenance of geometric associations and consistency*
- *Automated form feature generation*
- *Design for Manufacturability considerations: Feedback on Assembly/ Test time*
- *Manipulation of design features*
- *Inferencing using the CLIPS expert system shell*

462

This system was implemented at the case study site and was acknowledged to be essential to enhance the collaborative aspects of the design process.

Conclusions

This paper has stressed the need to augment the capabilities of computer based design aides to provide information based capabilities that allow them to integrate with the design processes. This is particularly significant in multidisciplinary design environments where the common thread linking various design entities is the common information generated by the different design agents. There is a need to provide information management capabilities at the source of the design action to externalize the information burden associated with such design activities. This paper has proposed a framework to integrate geometry based design systems with a centralized blackboard database model. Some of the essential components of this framework are briefly discussed and a brief outline of the capabilities of the prototype system is given. This system was implemented at the sponsor's design/manufacturing facility and was well received by the staff.

REFERENCES

[ALEX64]   Alexander, C.., (1964).   *Notes on the Synthesis of Form.*   Harvard University Press.

[AUTO89]   AutoCAD Reference Manual, Release 10. Autodesk, 1989.

[ORAC87]   *ORACLE DBMS* Reference Manual. ORACLE Corporation, California, 1987.

[PAHL84]   Pahl, G., and Beitz, W., (1984) *Engineering Design* . Design Council. London.

[RANG89]   Rangan, R., M., Fulton, R., E., & Woolsey, T., (1989).   EMTRIS Controlling Design and Manufacturing Information in a Discrete Part Manufacturing Environment. *ASME Computers in Engineering Conference, CA, 1989.*

[RANG90]   Rangan, R., M., Fulton, R., E., & Woolsey, T., (1989).   An Information Based CAD Design Environment to Support Multidisciplinary Design. *ASME Computers in Engineering Conference, CA,1989*

[RANG190]   Rangan, R. M., (1990). Enforcing Compatibility & Constraint Conditions and Information Retrieval at the Design Action.   CLIPS User's Group Conference, TX. August, 1990.

[SRIR89]   Sriram, D., et. al. (1989). "Knowledg-Based System Applications in Engineering Design: Research at MIT". *AI Magazine*, fall, 1989.

[STIN80]   Stiny, G., (1980). An Introduction to Shape and Shape Grammars. *Environment and Planning*, B, 7:343-351, 1980.

[TEOR86]   Teorey, T., J., Yang, D., Fry, J., P. , A Logical Methodology for Relational Database Design Using the Extended Entity-Relationship Model, *ACM Computing Survey*, Vol. 18, No. 2, June, 1986.

463

Aircraft Design Optimization
Using a Quasi-Procedural Method and Expert System

Ilan Kroo and Masami Takai
Department of Aeronautics and Astronautics, Stanford University
Stanford, California

Introduction

This paper describes a new program architecture for complex engineering design and illustrates its application to aircraft design optimization. This *quasi-procedural* method selects and executes the analysis subroutines for the calculation of objective function and constraints. Furthermore, it decides which variables need to be recomputed in response to the change of a design variable, permitting the objective and constraints to be recalculated efficiently. A rule-based expert system is also used to identify active constraints and suggest solutions to make the design feasible. The integrated optimizer, quasi-procedural program, and expert system are applied to the aerodynamic optimization of a swept wing and to the complete synthesis of a medium range commercial aircraft. The performance of the system is compared with that of conventional programs.

Quasi-Procedural Method

The quasi-procedural method (Ref. 1) is a form of nonprocedural program, consisting of a set of small compiled subroutines and an executive routine that keeps track of the subroutine and variable dependencies. In response to a request for the value of a certain variable, the executive program calls the relevant routines in the appropriate order. In the example shown in figure 1, the system traces the variable dependencies, through intermediate results of many subroutines, from the desired output to the input variables shown at the top of the figure. This real-time arrangement of the computational path offers improved flexibility and extensibility compared with conventional procedural methods. In addition, because the structure of the computations is known, the system performs consistency maintenance, recognizing, for example, that changes in the variable B have no effect on the desired output and therefore do not require recomputation. This improves the system efficiency and is particularly significant when the quasi-procedural analysis is combined with numerical optimization.
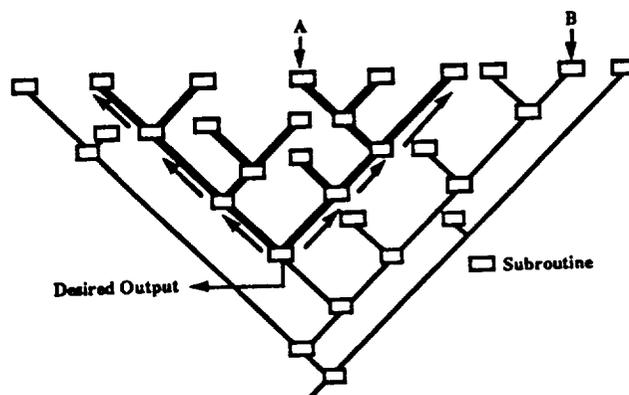
Figure 1. Quasi-Procedural Method and Consistency Maintenance

Numerical Optimization Using Quasi-Procedural Programming

Numerical optimization involves repeated evaluation of an objective function and constraints. At each evaluation, modification of some design variable (e.g., wing area) may require re-execution of a large number of analysis modules. On the other hand, changes in other variables, such as takeoff flap deflection, may invalidate only a few analyses. The quasi-procedural method recognizes the difference and re-configures the computational path, allowing the optimizer to avoid the redundant calculations made by conventional methods with rigid program structures. This improved efficiency may be utilized in several ways. The following sections describe three methods by which the quasi-procedural method may be used to improve the performance of conventional numerical optimization methods.

Gradient Calculation

Time savings are especially large when the gradient of the objective function must be computed by finite differences. In this case, each component of the gradient is constructed by evaluating the change in objective function due to a change in the corresponding design variable. Since this process involves changing only a single design variable at a time, much of the computational path is unaffected and so the number of required computations is reduced. Furthermore, if the calculation of one gradient component requires a time T, which is substantially longer than the times required to compute the other components in a N-dimensional optimization, a fixed program structure evaluates the gradient in a time N*T, while the quasi-procedural method demands only a bit longer than T.

Constrained Optimization

Constrained optimization problems provide additional opportunities for reduction in computation time. Figure 2 shows a feasible region bounded by five constraints. The gradients of the objective and constraint functions are evaluated at the end of each line search. The size of a finite difference interval for gradient calculation must be small enough to approximate the derivatives accurately. This means that if a constraint is inactive at a point, that constraint is not violated during the gradient calculation at that point. The quasi-procedural method makes it easy to remove inactive constraints from gradient calculations. The resulting computational time savings is large when expensive constraints are inactive during much of the search. Further savings are achieved because not all design variables affect all constraints. Since changing take-off flap deflection affects climb and take-off field length, but not range or landing field length, no additional time is spent computing range when take-off flap is varied. This makes it possible to achieve the efficiency available with a reduced design variable set without actually changing design variables.
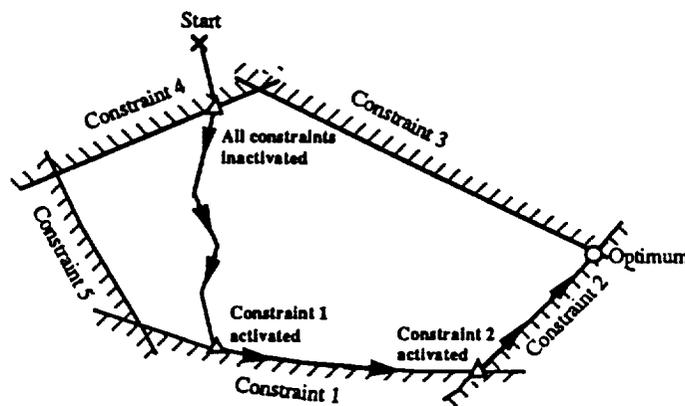
Figure 2. Constrained Optimization and Activation of Constriants

Use of the Chain Rule

The availability of subroutine dependency information, provided by the quasi-procedural method, enables the use of the chain rule, which may reduce the cost of derivative calculations (Ref. 2). Consider the analysis procedure shown in Figure 3. If the entire calculation were treated as a black box the time required for computation of the gradient of q with respect to the seven design variables would be:

$7*(A+B+C+D+E+F+G+H+I)$ where A, B, C, etc. represent the time required for the corresponding routine (or twice this value if central-differencing is used.) However, given the dependency information, the derivative of q with respect to a may be written:

$$\partial q/\partial a = \partial q/\partial o*\partial o/\partial a + \partial q/\partial p*\partial p/\partial a$$
$$= \partial q/\partial o*(\partial o/\partial l*\partial l/\partial a + \partial o/\partial m*\partial m/\partial a + \partial o/\partial n*\partial n/\partial a)$$
$$= \partial q/\partial o*(\partial o/\partial l*\partial l/\partial h*\partial h/\partial a + \partial o/\partial m*\partial m/\partial i*\partial i/\partial a)$$

Figure 3. Gradient Calculation by Chain Rule and Intermediate Sensitivities

This evaluation is performed in the time: $I+H+E+A+H+F$.
The entire gradient requires a time of:
$(I+H+E+A+H+F) + A + (B+G+H) + B + (C+G) + (D+I) + F$
$= 2A+2B+C+D+E+2F+2G+3H+2I$
Thus the evaluation time is reduced from 63 to 16 (in subroutine units). The savings become more significant when one of the subroutines which is called only once (i.e., C, D, or E) is very time-consuming. For example, if the routine E requires 10 times the computational effort as the other routines, the calculation time is reduced by 80%.

Expert System

Domain-specific knowledge can be especially useful in the early stages of aircraft design. Rather than start the optimization at an arbitrary point, an expert system is used to improve the initial design. This rule-based system was combined with the quasi-procedural program to warn the user of active constraints or other design problems, and to offer intelligent advice on how the problem might be corrected (Ref. 3).

The warning rules examine the current database, identify active design constraints, and report them to the user. A typical warning rule may compare the current value of a variable with its required value, and issue a warning string in case of constraint violation.

Solution rules analyze the causes of a constraint violation and generate solutions using design knowledge. A solution rule first looks at the warnings posted by the warning rules, and tries to identify a specific problem for which the solution rule is responsible. The rule may then look at the database to collect more information pertinent to the current problem and prescribe a solution best-suited for the current case. For example, a set of solution rules for the takeoff distance problem might be:

```
IF (TOFieldLength is too long) and not(ClimbGrad is too small)
and (*< TOFlapDefl 20.) THEN (Increase TOFlapDefl)


IF (TOFieldLength is too long) and (*/ TotalSLST MaxTOW $ToverW)
and (*< $ToverW 0.2) THEN (Increase SLSThrust)
```

A rule-base with approximately 100 rules was used to resolve fundamental problems with the initial design so that the numerical optimization could be started in a feasible region.

Applications

The quasi-procedural analysis method was combined with a variable-metric optimizer to illustrate the efficiency of the system in realistic design applications. In this section, two example problems are discussed: the design of a swept wing using linear potential theory, and an aircraft synthesis and sizing problem in which direct operating cost is minimized.

Wing Design

In this example, the linearity of an aerodynamic analysis routine is exploited by the quasi-procedural method and nonlinear optimizer. The wing twist distribution is to be designed so that the induced drag is kept low, the lift coefficient distribution is relatively uniform, and the desired wing lift is achieved. The design variables include the wing taper ratio, the twist angle of each of the twenty panels, and the angle of attack. A quasi-Newton optimizer (Ref. 4) was used to minimize the objective function with a central differencing scheme for gradient calculation.

Figure 4 shows the geometry and vortex arrangement of the swept wing. The spanwise lift distribution and induced drag are computed based on a Weissinger method, using a discrete vortex representation of the wake and a concentrated bound vortex. The trailing vortices are evenly spaced along the span, and the discrete bound vortices are placed at the quarter chord. Using the Biot-Savart law, an aerodynamic influence coefficient (AIC) matrix is computed which relates the strengths of the discrete vortices Γ to the downwash w at the control points located at the three quarter chord of the spanwise panels: $AIC(i,j) = w_i$ due to unit vortex strength at j The strengths of the bound vortices, and eventually the spanwise lift and C_l distributions, are then found by solving the linear equation: $[AIC]\{\Gamma\} = U_\infty\{\theta\}$ where U_∞ is the freestream velocity.
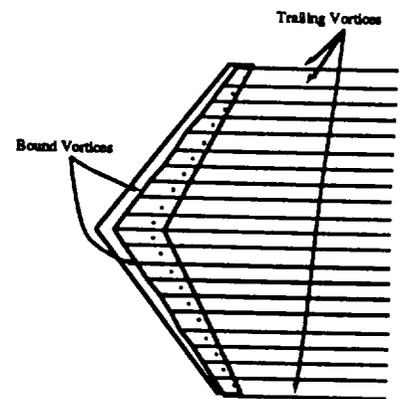
Figure 4. Wing Vortex Model

Figure 5 shows the program structure for the wing design problem. The subroutine AIC constructs the elements of the AIC matrix. Decomp performs LU decomposition of the AIC matrix, and Solve performs back-substitution. The AIC matrix has to be recomputed when taper ratio is modified, but need not be recalculated if only the twist angles and angle of attack are changed. The number shown to the right of each subroutine box in figure 5 is the execution time of the subroutine as a fraction of the total execution time. Solve is much faster than AIC, accounting for only 16% of the total computational time. This computational structure permits the quasi-procedural method to efficiently compute the gradient components with respect to the 20 twist angles and angle of attack. Figure 6 shows the optimal C_l and lift distributions. Note that the optimal C_l distribution is nearly constant as desired, and the lift distribution is nearly elliptic as reflected in the span efficiency of 0.98.
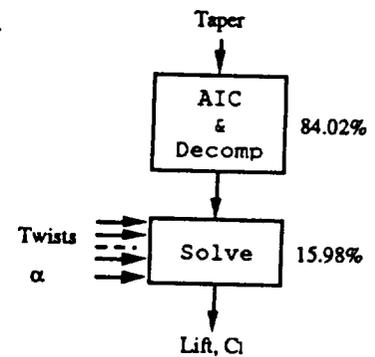
Figure 5. Structure of the Analysis Routines

Because a central differencing scheme is used, every gradient calculation requires evaluation of the AIC matrix twice each change in taper ratio, and solution of the linear system 42 times (2 times for taper ratio, and 40 times for twists). Therefore, the computational time for one gradient evaluation is: 2 * 5.33 sec + 42 * 1.01 sec = 53.2 sec. This figure may be compared with the time needed by conventional methods which do not recognize the structure of the program: 42 * (5.33 sec + 1.01 sec) = 266.3 sec. The ratio of these two values indicates a 80% saving by the quasi-procedural method.

467
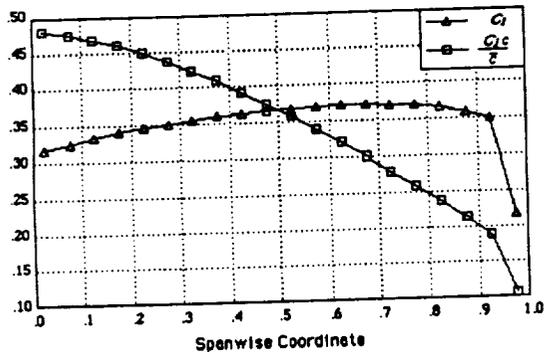
Figure 6. Optimal Lift and C_l Distributions
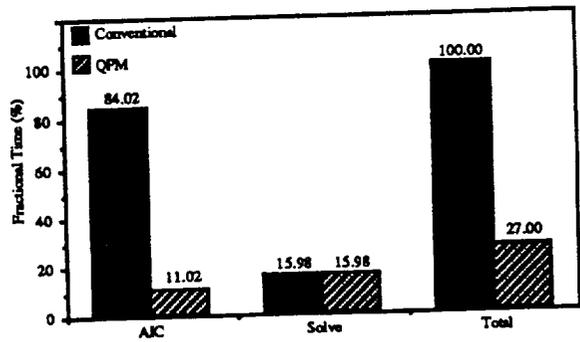
Figure 7. Computation Time Distribution

Figure 7 compares the total optimization times between the quasi-procedural and conventional methods. The figure shows the significant time saving obtained in the AIC calculation. The total computational saving is 73%, which is only 7% less than the maximum 80% gain available in the gradient calculation.

Complete Aircraft Synthesis

To illustrate the application of the method in a more complex problem, a set of aircraft analysis routines were written and used in the optimization of a medium range commercial aircraft. Ten design variables, shown in figure 8, were used to minimize the direct operating cost subject to constraints on range, landing and take-off field lengths, engine-out climb gradient, and cruise thrust. Figure 9 illustrates the major sub-routines required for the calculation and their relationships with the objective and constraint routines. The analysis routines include geometric, aerodynamic, structural, and economic computations based on preliminary design methods of Douglas Aircraft Company (Ref. 5) which were modified for this application.

The expert system was first employed to generate a reasonable starting point for the numerical optimization. In this case, the system was able to suggest solutions to provide a feasible starting point. The variable metric optimizer was then used to minimize direct operating cost. Figure 8 shows the converged solution which satisfies the five constraints and reduces DOC by 4% compared with the initial design.

Design Variables

MaxTOW	95739 lbs
WingArea	950.2 sq ft
SweepW	35.9°
ARw	7.12
Wing t+c	.146

Mach = 0.80
Range = 1500 n. mi
TOFieldLength < 6000 ft
LandFieldLength < 5500
ClimbGrad > 0.024

InitialCrAlt	29990 ft
FinalCrAlt	30046 ft
SLSThrust	16797 lbs
TOFlapDefl	12.4°
Sh+Swing	.273
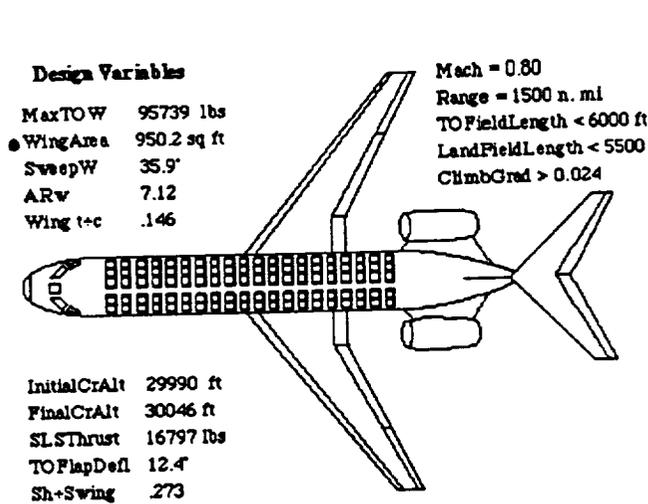
Figure 8. Optimal Geometry and Design Variables
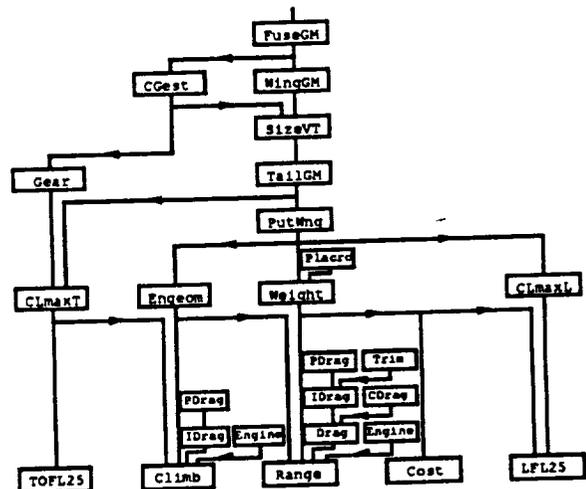
Figure 9. Computational Path for DOC and Constraints

468

Figure 10 shows the amount of time spent in each of the subroutines for three optimization cases. In the first case, the subroutines were arranged in a suitable order and the quasi-procedural system was not employed. In the second case, the system with consistency maintenance was used. Finally, inactive constraints were removed from gradient calculations as described previously. The result is an overall reduction in computation time of 22% for the quasi-procedural method, increasing to 39% when inactive constraints are removed. The figure shows that routines such as fuselage geometry (FUSEGM) are not affected by the selected design variables and so need to be executed only once.
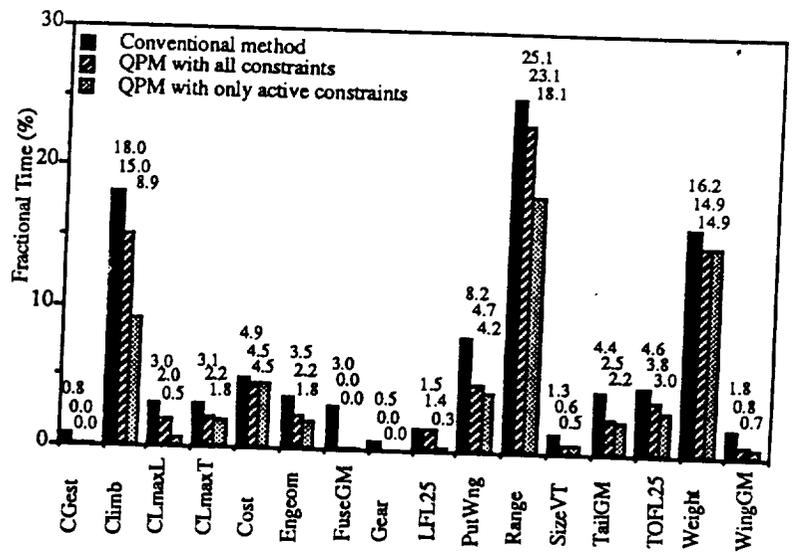
Figure 10. Distribution of Computation Times Among Analysis Routines

Conclusions and Continuing Work

The quasi-procedural method can significantly improve the performance of conventional numerical optimization. This is achieved primarily by the automatic simplification of the computational path during gradient calculations. In certain cases, savings of up to 80% in computation time are realized. Little improvement is seen during the line search portion of the optimization, however, since all variables are changed simultaneously at each step. Current work includes the investigation of non-gradient based methods (e.g. genetic algorithms) that may also be well-suited for use with quasi-procedural analysis.

References

1. Kroo, I., Takai, M., "A Quasi-Procedural, Knowledge-Based System for Aircraft Design," AIAA Aircraft Design, Systems and Operations Meeting, AIAA paper No. 88-4428, September 1988.

2. Sobieski, J. S., "Sensitivity of Complex, Internally Coupled Systems," AIAA Journal, Vol. 28, No. 1, p.153-160, January 1990.

3. Takai, M., A New Architecture and Expert System for Aircraft Design Synthesis, Stanford University, Ph.D. Thesis, June 1990.

4. Gill, P. E., Murray, W., Wright, M. H., Practical Optimization, Academic Press, London, 1981.

5. Kroo, I., "Tail Sizing for Fuel-Efficient Transports," AIAA paper No. 83-2476, October 1983.

An Application of Compound Scaling to Wind Tunnel Model Design

Mark French and Raymond M. Kolonay
Wright Research and Development Center
Wright-Patterson AFB, OH

Abstract

An approach has been developed for the stiffness design of aeroelastically scaled wind tunnel models. The object of designing such models is to make a structure whose stiffness matches a desired stiffness distribution. This design problem is cast as a formal constrained optimization problem and worked with two different optimization methods. A previous effort used the modified method of feasible directions (MFD) as implemented in a general purpose finite element based optimization code. In this effort, a special purpose finite element based optimization program was written and run using both MFD and compound scaling optimization methods. Results are presented comparing the final designs obtained using MFD and compound scaling.

Introduction

One of the most important tasks in the design of an aeroelastically scaled wind tunnel model wing is making sure the stiffness characteristics are correct. The engineer must design a model structure whose stiffness characteristics match known values. These known values are derived by scaling the stiffness characteristics of the full-size structure that the model represents.

It is assumed that a target flexibility matrix is known for the scaled model. A column of a flexibility matrix represents the displacements at all the grid points on the wing due to a unit load at one grid point. If the desired flexibility matrix is known for the model, a displacement constraint can be written for the deflection at each grid point in the structure due to a unit load placed at some grid point. If a structure can be designed so that all the constraints are active, it will have the desired response to the unit load placed on it. That is, if all the constraints written for a unit load corresponding to the nth degree of freedom are active, the nth column of the calculated model flexibility matrix will match the nth column of the desired flexibility matrix.

If this optimization approach works, there is no need to know any more about the desired structure than the flexibility matrix and the basic geometry of the structure. The model's structure need not bear any resemblance to that of the original structure as long as the overall stiffness characteristics match the scaled stiffness of the full-size structure. The possibility then exists of modelling an anisotropic structure with an isotropic one.

It has been shown that a general purpose structural optimization code can be used to simplify the wind tunnel model design process considerably (Ref 1). The previous approach was more cumbersome than desirable since the program involved (Ref 2) was applied to a problem for which it was not well suited. To improve the design procedure, a new finite element based optimization program was written and the problem described in Reference 1 was reworked.

Analysis

Before a model could be designed, a wing had to be selected for which stiffness characteristics were well known. To allow easy comparison of the results between the current research and that cited in Reference 1, the same 1/9 scale fighter wing was selected. The wing was constructed largely of

composite materials and was built to demonstrate the feasibility of using aeroelastic tailoring on fighter wings (See Figure 1). The structure involved in this effort represents only the structural box of the original wing. The model structure described here is the same size as the 1/9 scale wing.

Reference 3 presents a flexibility matrix generated from a finite element model of the 1/9 scale wing. Flexibility coefficients are given at 28 points on the surface of the wing. The location of these points is presented in Figure 2. This flexibility matrix was used as the basis for the design of the test structure. A set of displacement constraints for the optimization problem can be written using a column of the flexibility matrix; a load is placed at some point on the wing and the terms from the column of the flexibility matrix are input as displacement constraints. Each column of the flexibility matrix is considered as a separate load case. In vector form, a normalized set of constraints can be written as

$$g_j(x) = \frac{x_{ij} - \delta_{ij}}{\delta_{ij}} \leq 0, \quad j = 1, n \tag{1}$$

where x_{ij} is the calculated displacement at the jth grid point due to a unit load at the ith grid point and δ_{ij} is the desired displacement at the jth grid point due to a unit load at the ith grid point.

Previously, structural mass was used as the objective function since it is assumed in the code used. This choice is reasonable since one would expect a minimum weight structure to be a relatively flexible one and thus one for which many of the constraints are active or nearly so. However, since the goal of the design procedure is to minimize the difference between the desired and actual deformation at the grid points, the problem was reformulated to use a squared error function as the objective function.

$$F = \sum_{i=1}^{n} \left(\frac{\delta_i - x_i}{\delta_i} \right)^2 = \sum_{i=1}^{n} g_i(x)^2 \tag{2}$$

The design variables for the problem are the widths and heights of the beam elements. The values do not show up explicity in the objective function; however, the calculated deformations which are used in forming the objective function are in turn functions of the element sizes. The formal optimization problem can be stated as

$$\begin{aligned}
\text{Minimize:} \quad & F(x(h,b)) \\
\text{Subject to:} \quad & g_i \leq \bar{g}_i \\
& b_i^{min} \leq b_i \leq b_i^{max} \\
& h_i^{min} \leq h_i \leq h_i^{max}
\end{aligned} \tag{3}$$

The 1/9 scale wing model chosen was originally intended for transonic testing and is extremely stiff. To make stiffness testing of the new model easier, the terms of the flexibility matrix were multiplied by 10. This was considered reasonable since the purpose of this experiment is to show that an arbitrary stiffness distribution can be modelled with a simple isotropic structure, rather than model a specific wing. The result was a structure flexible enough to give easily measurable displacements under a modest load.

In order to exactly match the scaled stiffness properties of the 1/9 scale model, a complete set of displacement constraints would need to be applied for every column of the flexibility matrix. However, this approach would result in an unreasonably large problem. Another approach is to appeal to intuition and assume that if columns of the flexibility matrix are correct for a small number of unit loads applied at widely separated grid points, the remaining columns are correct or nearly so.

A second assumption can be made to further simplify the optimization process. The deformations at the inboard grid points are often very small but can have undesireably large influence on the design problem; very small absolute differences between the desired and calculated displacements can still result in very large normalized constraint values. Ignoring constraints at the inboard grid points when calculating the objective function helps ensure that the optimization algorithm does not get bogged down trying to cope with highly violated constraints which have little effect on the results. All constraints, including those omitted from the objective function calculation were used to define the boundary of the feasible region.

Previously, a structure composed of beam and plate bending elements was used to represent the structure of the aeroelastically scaled model. The results were good, but the existence of the plate elements caused problems in fabricating a test specimen. To address this problem, only beam elements were used in this effort. The structure is a lattice of beam elements connecting the points at which flexibility coefficients were given.

A FORTRAN program was written to model the wing structure using bar finite elements and generate function values and gradients for passing to an optimization program. The program uses 12 DOF beam elements to model the bars. The finite element model is presented in Figure 2. Two different optimization methods were used for different versions of the program: ADS (Ref 4) and compound scaling as implemented in FUNOPT (FUNctional OPTimization, Refs 5 and 6). The Modified Method of Feasible Directions (MFD) was used in ADS.

A number of different constraint sets was specified during the course of the design process. Initial runs were made using a single set of constraints and the number was gradually increased. It was found that four displacement vectors were sufficient to design the wing. The four columns of the flexibility matrix correspond to unit loads placed at the leading edge of the tip, trailing edge of the tip, leading edge of the mid span and trailing edge of the mid span. These correspond to points 13, 16, 25 and 28 on Figure 2. The resulting design problem had 106 design variables (width and height for each of 53 beam elements) and 112 constraints (4 load cases and 28 constraints per load case).

Results

Both methods produced acceptable final designs. The ADS design had 28 constraints within 5% of the boundary of the feasible region, but most of these were concentrated in the last load case. For the second load case, there were no constraints within 5% of the boundary. The FUNOPT design was the better of the two since approximately 60 constraints were within 5% of the boundary of the feasible region. This indicates that FUNOPT was able to find an approximate intersection of 60 constraints. The caveat should be made that both methods used here have parameters which can be varied to improve performance. ADS may have produced a better design if the optimzation parameters had been set differently.

It is difficult to compactly present results from the analyses described here, but some feel for the quality of the results can be obtained by examining the deflections along the leading and trailing edges of the

472

structure due to a unit load at a tip grid point. Figure 3 compares the displacements along the leading edges of the ADS and FUNOPT designs due to a unit load at grid point 28 to the desired displacements. Figure 4 compares the displacements at the trailing edge of the two designs with the desired displacements due to the same load.

The biggest problem encountered in the optimization process was getting the optimization routines to converge on an answer. ADS in particular was very sensitive to the initial design chosen. If the initial design was infeasible, ADS often could not find a feasible solution. The strategy which seemed to work the best with ADS was to vary the initial design by hand until the none of the constraints were violated. There was very little trouble getting FUNOPT to converge to a good answer; however, there is no good exit criteria yet. The result is that it will often run longer than desired.

Conclusions

It is difficult to determine how much deviation between actual and desired stiffness properties is allowable in aeroelasticity models. (Ideally there is none, but this is almost never the case.) The allowable deviation depends on such factors as wing geometry and the flight regime. The results presented above indicate that using structural optimization to design wind tunnel models can result in a procedure which matches desired stiffnesses well enough to be very useful in sizing the structures of aeroelastic models.

The design procedure presented here demonstrates that optimization can be useful in designing aeroelastically scaled wind tunnel models. The resulting structure effectively models an aeroelastically tailored composite wing with a simple aluminum beam structure. This structure should be inexpensive to manufacture compared to a composite one. Furthermore, it should be easier to manufacture than the structure designed in the previous effort (Ref 1).

References

1. French,R.M., "An Application of Structural Optimization in Wind Tunnel Model Design", AIAA-90-0956-CP, April 1990.

2. Neill, D.J., Johnson, E.H. and Canfield, R.A., "ASTROS - A Multidisciplinary Automated Structural Design Tool", AIAA-87-0713, April 1987.

3. Rogers, W.A., Braymen, W.W., Murphy, A.C., Graham, D.H. and Love, M.H., "Validation of Aeroelastic Tailoring by Static Aeroelastic and Flutter Tests", AFWAL-TR-81-3160, September 1981.

4. Vanderplaats, G.N., "An Efficient Feasible Directions Algorithm for Design Synthesis", AIAA Journal, Volume 22, Number 11, November 1984, pp 1633-1640.

5. Venkayya, V.B., Tischler, V.A., "A Compound Scaling Algorithm for Mathematical Optimization", WRDC-TR-89-3040, February 1990.

6. Venkayya, V.B., Tischler, V.A., Kolonay, R.M., Canfield, R.A., "A Generalized Optimality Criteria for Mathematical Optimization", AIAA 90-1192, April 1990.
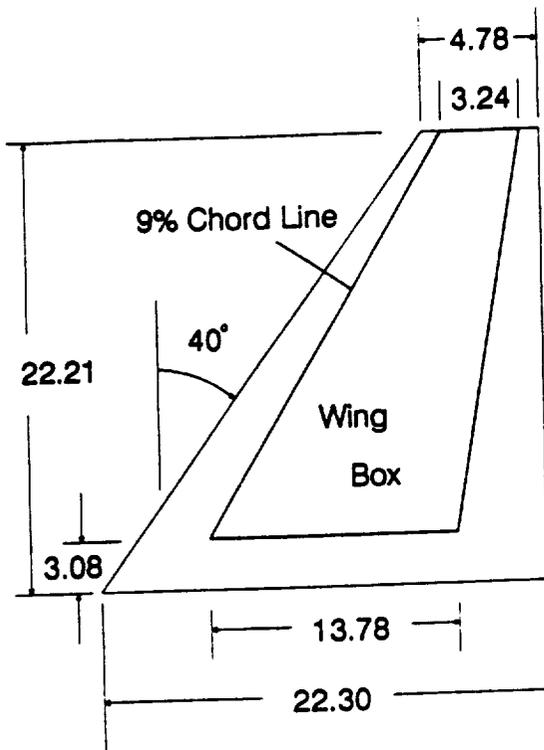
Figure 1. Planform of 1/9 Scale Wing
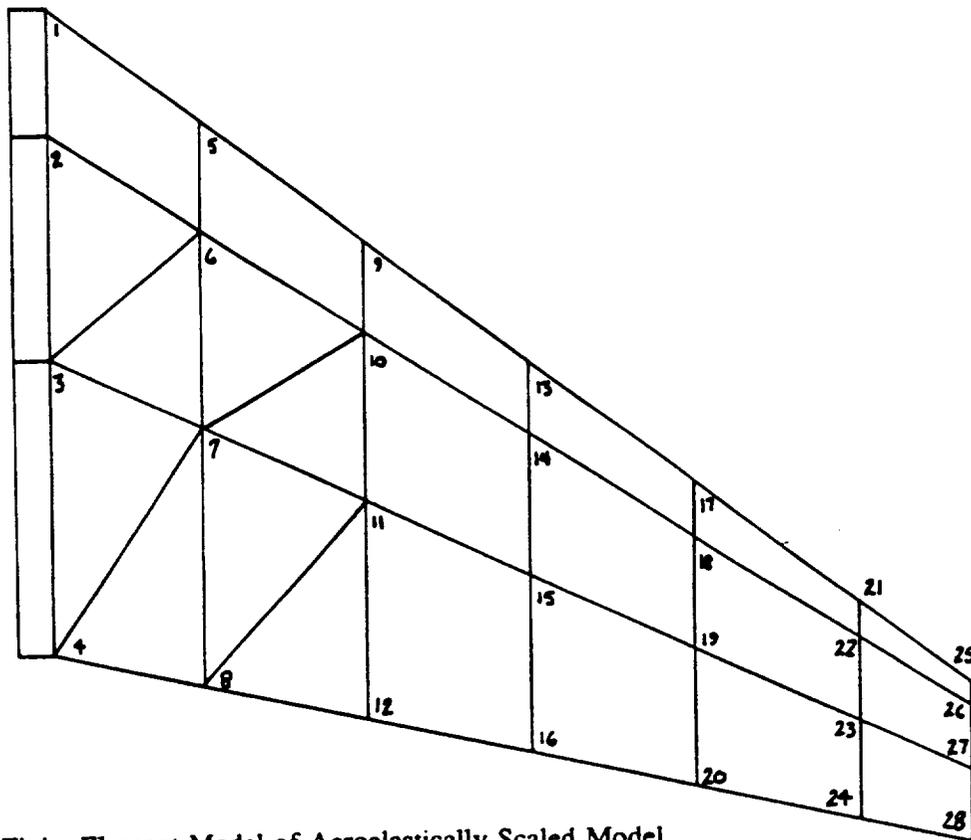
Figure 2. Finite Element Model of Aeroelastically Scaled Model.
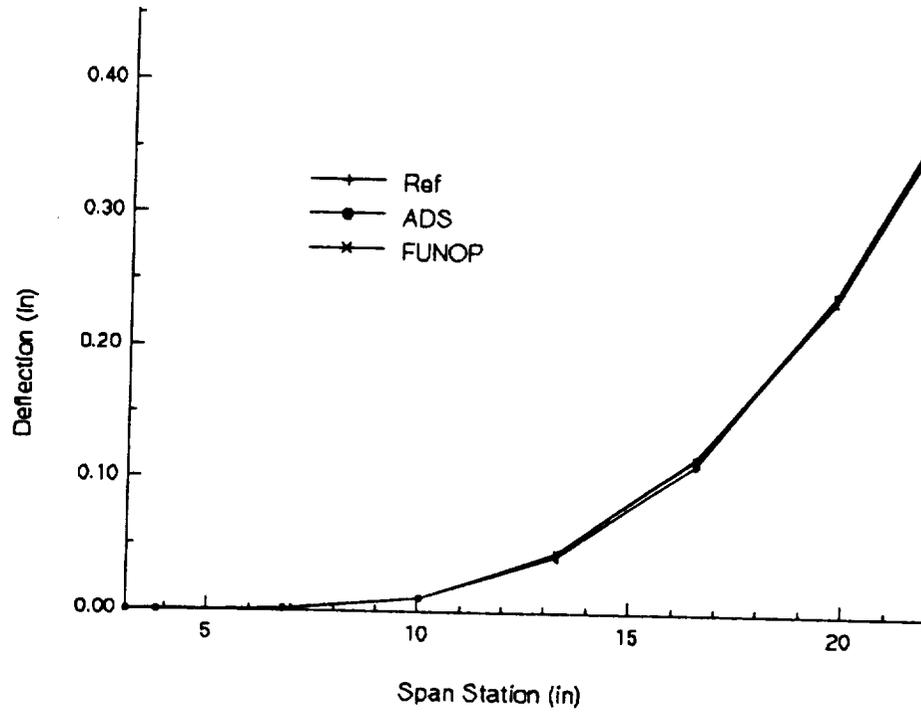
Figure 3. Deflections Along LE Due to Unit Load at Pt 28.
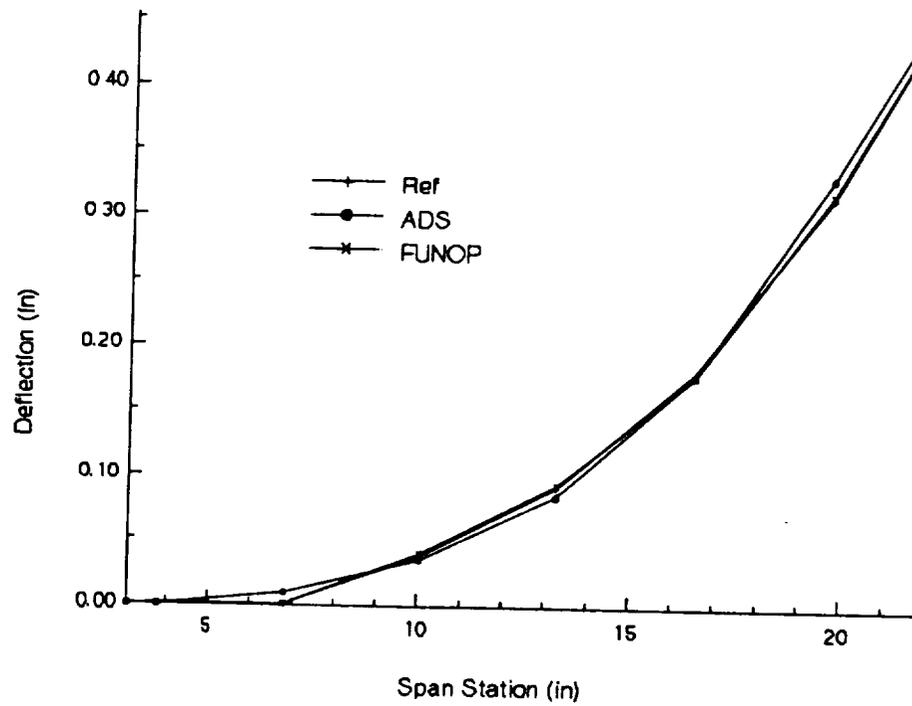
Figure 4. Deflections Along TE Due to Unit Load at Pt 28.

HITCAN: HIGH TEMPERATURE COMPOSITE ANALYZER

Surendra N. Singhal[1], Joseph J. Lackney[1],
Christos C. Chamis[2], and Pappu L. N. Murthy[2]

[1]Sverdrup Technology, Inc., Brook Park, Ohio
[2]Lewis Research Center, Cleveland, Ohio

ABSTRACT

A computer code, HITCAN (HIgh Temperature Composite ANalyzer) has been developed to analyze/design metal matrix composite structures. HITCAN is based on composite mechanics theories and computer codes developed at the National Aeronautics and Space Administration (NASA), Lewis Research Center, over the last two decades. HITCAN is a general purpose code for predicting the global structural and local stress-strain response of multilayered (arbitrarily oriented) metal matrix structures both at the constituent (fiber, matrix, and interphase) and the structure level and including the fabrication process effects. The thermo-mechanical properties of the constituents are considered to be nonlinearly dependent on several parameters including temperature, stress, and stress rate. The computational procedure employs an incremental iterative nonlinear approach utilizing a multifactor-interaction material behavior model. HITCAN features and analysis capabilities (static, load stepping, modal, and buckling) are demonstrated through typical example problems.

BACKGROUND

High temperature metal matrix composites (HTMMC) have shown potential as structural materials for 21st century propulsion systems. The nonlinear dependence of the thermo-mechanical properties of HTMMCs on parameters such as temperature, stress, and stress rate, may alter the structural response significantly. Experimental investigations being high in cost, computational models including nonlinear material behavior simulating the real-life response of components made from HTMMC materials are required.

The need for developing multilevel analysis models for multilayered fibrous composites was recognized almost 2 decades ago (Ref. 1) and a multilevel analysis computer code was developed subsequently (Ref. 2). Research related to various aspects of HTMMC materials and structures has been conducted at the Lewis Research Center of the National Aeronautics and Space Administration (NASA) for several years. Building upon parts of this research effort, a high temperature composites analyzer code HITCAN, has been developed.

The objective of this paper is to summarize HITCAN's capabilities with illustrative examples which demonstrate its application versatility.

HITCAN: A BRIEF DESCRIPTION

HITCAN presents a self-contained (independent of commercial codes) synergistic combination of NASA developed codes, MHOST and METCAN which are finite element structural analysis, and multilevel nonlinear material behavior

codes, respectively. User-friendliness was kept in mind during the development of HITCAN code. For instance, it includes a material property database for commonly used aerospace fiber and matrix materials. Properties from this data base can be input to HITCAN automatically.

Figure 1 shows the approach used by HITCAN for analyzing composite structures. The left part of Figure 1 depicts the determination of laminate properties based on known constituent properties. The top part depicts the finite element analysis which provides the structural response at the laminate level. And, the right part shows the determination of the structural response at the constituent level. Finally, the bottom left part shows the updating of constituent material properties based on input parameters and calculated constituent stress response. Figure 2 shows various regions in which the constituent material properties are updated.

FEATURES

HITCAN is capable of predicting global structural and local stress-strain response of multilayered high temperature metal matrix composite structures both at the constituent (fiber, matrix, and interphase) and the structure level. HITCAN's analysis capabilities include static, load stepping, modal and buckling response. HITCAN treats thermo-mechanical properties at the constituent level as nonlinear multifactor-interaction functions of temperature, stress, and stress rate, as shown below (Ref. 3).

$$\frac{P}{P_o} = \left[\frac{T_M - T}{T_M - T_o}\right]^n \left[\frac{S_F - \sigma}{S_F - \sigma_o}\right]^m \left[\frac{\dot{S}_F - \dot{\sigma}_o}{\dot{S}_F - \dot{\sigma}}\right]^l$$

Where P and P_o denote current and reference condition material properties; T_M, T, and T_o denote melting, current, and reference temperature; S_F, σ, and σ_o denote fracture at T_o, current, and reference stress; $\dot{S}_F$, $\dot{\sigma}$, and $\dot{\sigma}_o$ denote appropriately selected, current, and reference stress rate, respectively. The exponents n, m, and l are empirical constants.

HITCAN includes the dependence of the behavior of the metal matrix composites on fabrication process variables and properties of an interphase between the fiber and matrix.

Table I lists HITCAN features. The features that have been demonstrated through example problems (Ref. 4) are marked 'tested' in Table I. Although the enhancement of the code continues, in its current form, it is applicable to a wide variety of composite structural analysis problems.

DEMONSTRATION PROBLEMS

The static, load stepping, modal, and buckling analysis capabilities of HITCAN are demonstrated for a simply supported plate (Figure 3) made of Si C/Ti-15-3-3. For all demonstration problems discussed in this paper, the laminate consists of 4 layers with a top-to-bottom ply lay-up of (0/±45/90), a fiber

477

volume ratio of 0.4, and a reference temperature of 70 °F. Also, demonstrated is the sensitivity of structural response with respect to various forms of multifactor-interaction constitutive models, various ply orientations, fiber degradation, and fabrication-induced stresses.

Since, the purpose of the present report is to demonstrate the capabilities of HITCAN rather than to provide benchmark results, a detailed discussion of the results is not included. Also, due to the unavailability of results in open literature for the complex problems modeled by HITCAN, it has not been possible to provide comparisons. However, the code has been verified for some classical problems.

STATIC ANALYSIS: The static analysis was conducted for a combined thermal and mechanical load similar to that in Figure 3, but applied in one step. The results for static analysis, being similar to the constant material property case of the load stepping analysis which follows, are not included here.

LOAD STEPPING ANALYSIS: The load stepping analysis is essentially a piecewise linear analysis where the load is applied incrementally in several steps with material properties updated at the end of each load step. The analysis was first performed for the base case with no fiber degradation, no fabrication, (0/±45/90) ply lay-up, and the most general form of the constitutive model described above in the 'FEATURES' section. The geometry, loading, and boundary conditions are shown in Figure 3. The base case results including displacements and constituent and ply level stresses are also shown in Figure 3. The displacements are in the global structural coordinate system and the stresses in the local materials coordinate system. Both of these coordinate systems are shown in Figure 1. Letters A, B, and C, used in Figure 3 are for various regions of constituent material nonuniformity, defined in Figure 2.

The load stepping analysis was repeated by varying one parameter at a time, for (i) four cases of constitutive models, namely, the constant material property case, material properties dependent on temperature only, material properties dependent on stress only, and material properties dependent on stress rate only, (ii) two cases of ply orientations, namely, the symmetric orientation of $(0/45)_s$, and balanced orientation of $(0/90)_s$, (iii) one case of fiber degradation by a factor of one-tenth of its original diameter, creating an interphase between the fiber and matrix with properties of the interphase taken as an average of the fiber and matrix properties, and (iv) one case of fabrication thermal loading (Figure 4) applied before the combined thermal and mechanical loading. The corresponding displacement and ply stress results for all these cases are tabulated in Figure 4. For the sake of brevity, the sensitivity analysis response is compared with the base case at the end of the third load step only. The effect of stress rate was found negligible in all cases, due to the very nature of the problems chosen. This effect will show up in the transient analysis.

The effect of using different forms of constitutive models was analyzed further, as shown in Figure 4. This Figure shows the importance of using material behavior models which are dependent on applied temperature and calculated stress response. Notice that the vertical axis of Figure 4 shows percentage increase in the displacement due to a change in the form of the

478

constitutive model, i.e. the increase in the displacement caused by the degradation of material properties according to the multi-factor interaction model. The increase in the displacement is measured from the case when material properties are considered constant. In Figure 4, the label 'Temperature Effect' refers to percentage increase in the displacement when the material properties are made temperature dependent only. Similarly, the label 'Stress Effect' refers to percentage increase in the displacement when the material properties are made stress dependent only. And, the label 'Combined Effect' refers to percentage increase in the displacement when the material properties are made dependent on temperature and stress simultaneously.

MODAL ANALYSIS: The modal analysis was performed for the same combined thermal and mechanical load (Figure 3) as that used in the 'Load Stepping Analysis'. Four modes were calculated (the code is capable of calculating as many modes as desired). The results for natural frequencies are shown in Figure 5.

BUCKLING ANALYSIS: The buckling analysis was first conducted for mechanical loading only. The first buckling mode was calculated (the code is capable of calculating as many modes as desired). The analysis was then repeated for two cases; for mechanical loading including fiber degradation and for combined thermo-mechanical loading without fiber degradation. For the first case, the fiber was degraded by a factor of one-tenth of its original diameter. The loadings, boundary conditions, and results are shown in Figure 6.

CONCLUSIONS

Several features and analysis capabilities of the high temperature composite analyzer code, HITCAN, have been demonstrated through example problems. These features make HITCAN a powerful, cost-effective tool for analyzing/designing metal matrix composite structures and components. Because of the multilevel analysis approach, HITCAN has the utility for studying the influence of individual constituent in-situ behavior on global structural response. HITCAN will help in material selection for specific applications, in analyzing sensitivity of structural response to various system parameters, and in providing structural response at all levels of material constituents.

REFERENCES

1. Chamis, C.C., "Design Oriented Analysis and Synthesis of Multilayered-Filamentary Structural Panels", Ph.D. thesis, Case Western Reserve University, Cleveland, Ohio, 1967.

2. Chamis, C.C., "Computer Code for the Analysis of Multilayered Fiber Composites-User's Manual," NASA TN-D-7013, 1971.

3. Hopkins, D.A. and Chamis, C.C., "A Unique Set of Mircromechanics Equations for High Temperature Metal Matrix Composites," NASA TM-87154, 1985.

4. Singhal, S. N., Lackney, J. J., Chamis, C.C., and Murthy, P.L.N., "Demonstration of Capabilities of High Temperature Composites Analyzer Code HITCAN," NASA TM-102560, 1989.

Table I. - HITCAN Capabilities for Composite Materials

Type of Analysis ↓ \ Type of Structure →	Beam	Plate	Ring	Curved Panel	Built-up Structure
Static	tested	tested	tested	tested	tested
Buckling	tested	tested	tested	tested	tested
Load Stepping	tested	tested	tested	tested	tested
Modal (Natural Vibration Modes)	tested	tested	tested	tested	tested
Time-domain	-	-	-	-	-
Loading					
Mechanical	tested	tested	tested	tested	tested
Thermal	tested	tested	tested	tested	tested
Cyclic	-	-	-	-	-
Impact	-	-	-	-	-
Constitutive Models					
$P = $ Constant	tested	tested	tested	tested	tested
$P = f(T)$ (temperature dependence)	tested	tested	tested	tested	tested
$P = f(\sigma)$ (stress dependence)	tested	tested	tested	tested	tested
$P = f(\dot{\sigma})$ (stress rate dependence)	tested	tested	tested	tested	tested
$P = f(t)$ (creep)	-	-	-	-	-
$P = f(T, \sigma, \dot{\sigma})$ (combination)	tested	tested	tested	tested	tested
$P = f(T, \sigma, \dot{\sigma}, t)$ (creep combination)	-	-	-	-	-
Fiber Degradation	tested	tested	tested	tested	tested
Fabrication-induced Stresses	tested	tested	tested	tested	tested
Ply Orientations					
Arbitrary	tested	tested	tested	tested	tested

Notation: P: Material properties   T: Temperature σ: Stress $\dot{\sigma}$: Stress rate   t: Time
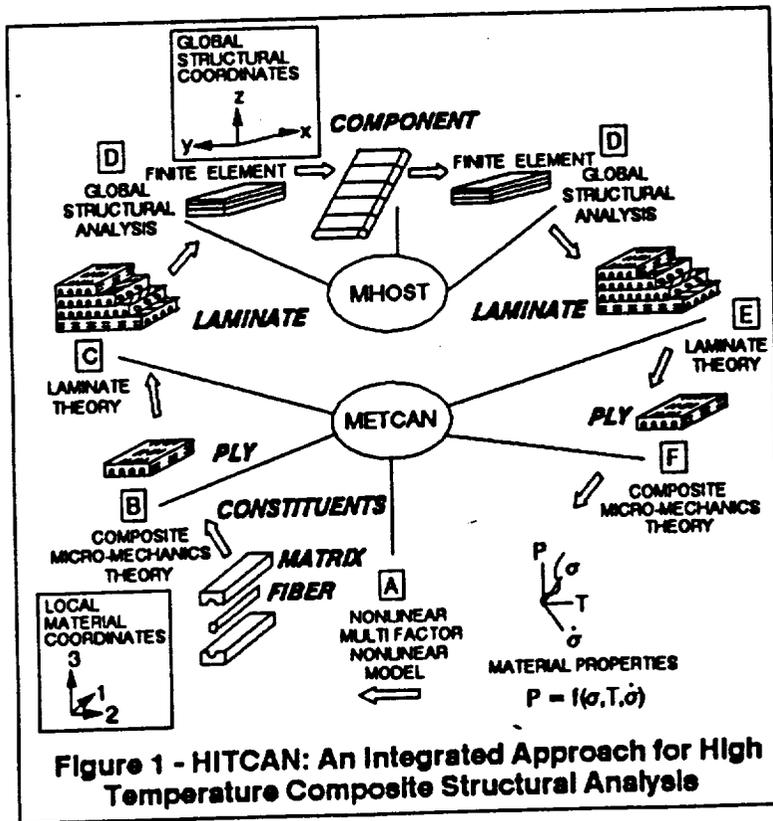
Figure 1 - HITCAN: An Integrated Approach for High Temperature Composite Structural Analysis

A: Matrix
B: Matrix and Interphase
C: Matrix, Fiber, and Interphase
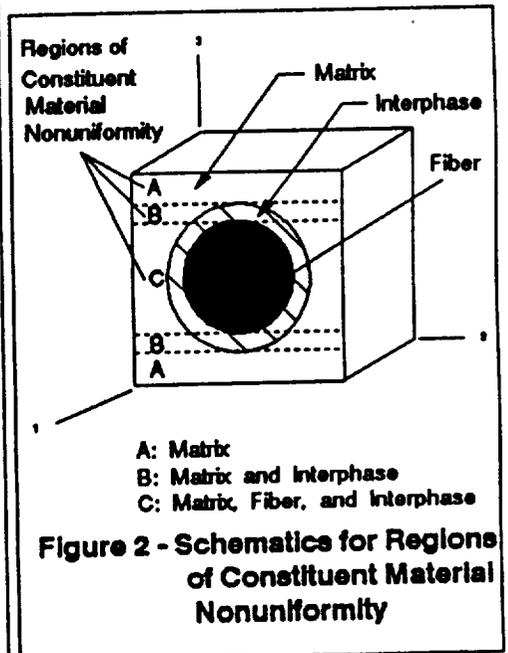
Figure 2 - Schematics for Regions of Constituent Material Nonuniformity

480

FIGURE 3 - LOAD STEPPING ANALYSIS: BASE CASE
SIMPLY SUPPORTED PLATE; SI C/TI-15-3-3-3 COMPOSITE; 0/±45/90 LAY-UP; 0.4 FVR

GEOMETRY

Z, Y, X
FORCE
TEMPERATURE
TEMPERATURE
6'
4'
0.125'

LOADING

FORCE (lb): 0, 50, 100, 150, 200
TEMPERATURE (°F): 0, 250, 500, 750, 1000
LOAD STEPS: 0 1 2 3 4
FORCE
TEMPERATURE

DISPLACEMENTS, CENTER POINT, Z DIRECTION

DISPLACEMENT (inch): 0, -0.004, -0.008, -0.012, -0.016
LOAD STEPS: 0 1 2 3 4

STRESSES, CENTER POINT, PLY 4 (BOTTOM PLY)

LONGITUDINAL STRESS (ksi): -20, 5, 30, 55, 80
LOAD STEPS: 0 1 2 3 4
FIBER, PLY, INTERPHASE(B), MATRIX(A)

TRANSVERSE STRESS (ksi): -30, -15, 0, 15, 30
LOAD STEPS: 0 1 2 3 4
FIBER, MATRIX(C), PLY, INTERPHASE(C), INTERPHASE(B), MATRIX(B), MATRIX(A)

SHEAR STRESS (ksi): -4, 0, 4, 8, 12
LOAD STEPS: 0 1 2 3 4
FIBER, MATRIX(C), MATRIX(B), PLY, INTERPHASE(B), MATRIX(A), INTERPHASE(C)

FIGURE 4 - LOAD STEPPING ANALYSIS: SENSITIVITY CASES

SENSITIVITY ANALYSIS AT LOAD STEP 3

EFFECTS OF	DISP. [CENTER POINT] (inch)	STRESSES, PLY 4 [CENTER POINT] (ksi)		
		LONG.	TRANS.	SHEAR
BASE CASE	-0.0135	23.9	0.8	1.3
NONLINEAR MULTIFACTOR INTERACTION CONSTITUTIVE MODELS				
ROPERTYP = CONSTANT	-0.0119	22.7	2.0	1.1
PROPERTY = f (TEMP.)	-0.0133	25.5	0.9	1.3
PROPERTY = f (STRESS)	-0.0121	21.5	1.9	1.1
PROPERTY = f (STRESS RATE)	-0.0118	22.7	2.0	1.1
PLY ORIENTATIONS				
(0/45) s	-0.0144	11.3	7.1	-1.6
(0/90) s	-0.0149	13.5	4.7	0.0
FIBER DEGRADATION	-0.0151	21.8	1.4	1.1
FABRICATION	-0.0140	18.4	5.8	0.4

FABRICATION COOLING LOAD

TEMPERATURE (°F): 0, 250, 500, 750, 1000
LOAD STEPS: 0 1 2 3 4 5 6
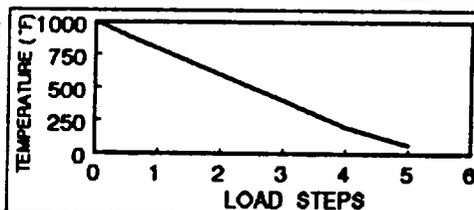
SENSITIVITY TO CONSTITUTIVE MODELS

PERCENTAGE INCREASE IN DISPLACEMENT: 0, 5, 10, 15
LOAD STEPS: 0 1 2 3
TEMPERATURE (°F): 70, 380, 690, 1000
Combined Effect, Stress Effect, Temperature Effect

FIGURE 5 - MODAL ANALYSIS

NATURAL FREQUENCY (cps): 0, 1500, 3000, 4500, 6000
LOAD STEPS: 0 1 2 3 4
4, 3, 2, MODE 1
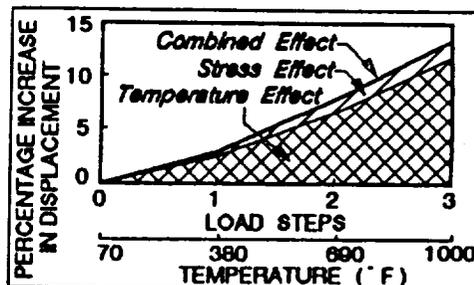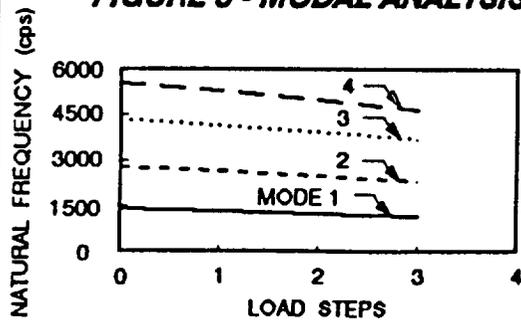
FIGURE 6 - BUCKLING ANALYSIS

AXIAL FORCE (25 lb/inch)
TEMPERATURE (1000°F)
TEMPERATURE (1000°F)
AXIAL FORCE (25 lb/inch)
6'
4'
0.125'
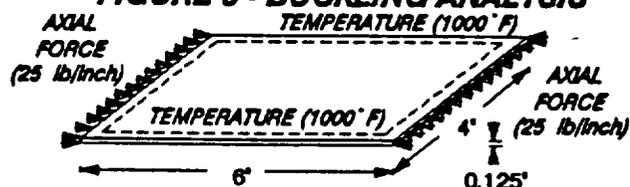
CRITICAL BUCKLING FORCE

(i) UNDER MECHANICAL LOADING ONLY = 939 lb/inch
(ii) WITH FIBER DEGRADATION, UNDER MECHANICAL LOADING ONLY = 901 lb/inch
(iii) UNDER THERMO-MECHANICAL LOADING = 675 lb/inch

A STUDY OF DELAMINATION BUCKLING OF LAMINATES

Yu Xie Mukherjee[1]   Xie Zhicheng[2]   Anthony Ingraffea[3]

1   Department of Theoretical and Applied Mechanics, Cornell University, Ithaca, NY, USA
2   Department of Engineering Mechanics,Tsinghua University, Beijing, PR China
3   Department of Civil and Environmental Engineering,Cornell University, Ithaca, NY, USA

Abstract

The subject of this paper is the buckling of laminated plates, with a pre-existing delamination, subjected to inplane loading. Each laminate is modelled as an orthotropic Mindlin plate. The analysis is carried out by a combination of the finite element and asymptotic expansion methods. By applying the finite element method, plates with general delamination regions can be studied. The asymptotic expansion method reduces the number of unknown variables of the eigenvalue equation to that of the equation for a single Kirchhoff plate. Numerical results are presented for several examples. The effects of the shape, size and position of the delamination, on the buckling load, are studied through these examples.

1   Introduction

A separated region might exist between the layers of a laminated plate or shell [1,2,3]. Such a region is called a delamination. When a plate with delamination buckles, the critical load is lower than that for the plate without delamination. The separated portion will usually open when delamination buckling occurs and the delamination may grow due to the buckling[4,5].

Reference [1] is one of the earlier papers on delamination buckling. The delamination buckling of beams and the growth of delaminations have been studied by using the classical beam theory. Axisymmetric delamination buckling has been studied in reference [6]. The buckling of a plate with a rectangular delamination has been solved in reference [7]. However, in both reference [6] and reference [7], the buckling load is assumed to be equal to the value for the delaminated part, which is considered to be a clamped plate. Obviously, the solution can only be applied in some special cases.

In this paper, the delamination buckling of laminated plates is studied by applying the finite element method, so that we can study plates with general delamination regions. Each layer is modelled as an orthotropic Mindlin plate[8]. This model describes the mechanical behavior of composite plates in an accurate manner. The shearing strains in each layer are chosen as independent variables. In this way, the locking problem can be avoided[9]. Also, this approach enables us to apply the asymptotic expansion method, which reduces the number of unknown variables of the eigenvalue equation to that of the equation for a single Kirchhoff plate.

Buckling loads are calculated numerically for several example problems. The effects of the size, shape and position of the delamination on the in-plane buckling load for a laminated plate are studied in the paper. Also, the results show that, as expected, the effect of the material properties on the buckling load is significant.

2   The Governing Equations

Consider a N-layer laminated plate with a delamination, shown in Figure 1. Take the plane with the delamination as the x-y plane. There are N^U layers with z>0 and N^L layers with z<0, which are called

the upper part and the lower part of the plate, respectively.

The linear discrete layer model is used, i.e., each layer is considered as a Mindlin plate. For the ith layer ($i=1,2,\ldots N^U, 1,2,\ldots N^L$), the displacements are:

$$u^i(x,y,z) = u_{i-1}(x,y) + \theta_i(x,y)\,(z_i + \tfrac{1}{2}t_i)$$

$$= u_0(x,y) + \sum_{m=1}^{i} t_m^* \gamma_{xm} - h_i\,w_x + z_i\,(\gamma_{xi} - w_x) \tag{1}$$

$$v^i(x,y,z) = v_{i-1}(x,y) + \phi_i(x,y)\,(z_i + \tfrac{1}{2}t_i)$$

$$= v_0(x,y) + \sum_{m=1}^{i} t_m^* \gamma_{ym} - h_i\,w_y + z_i\,(\gamma_{yi} - w_y) \tag{2}$$

$$w^i(x,y,z) = w(x,y) \tag{3}$$

where $|t_i|$ is the thickness of the ith laminate, and

$$t_m^* = \begin{cases} t_m & m < i \\[2mm] \tfrac{1}{2}t_m & m = i \end{cases} \qquad\qquad h_i = \sum_{m=1}^{i} t_m^*$$

Here, u_{i-1}, v_{i-1} are the in-plane displacements of the lower surface when the ith layer belongs to the upper part, and the in-plane displacements of the upper surface when the ith layer belongs to the lower part. The displacements u_0 and v_0 are the in-plane displacements of the x-y plane. γ_{xi} and γ_{yi} are the shearing strains γ_{xz} and γ_{yz} of the ith layer.

Each layer is an orthortropic plate. The angle between the major principal material axis and the x-axis is a^i for the ith layer. The equations relating stresses and strains are

$$\left\{\begin{matrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{matrix}\right\}_i = [D_e]_i \left\{\begin{matrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{matrix}\right\}_i \qquad\qquad \left\{\begin{matrix} \tau_{xz} \\ \tau_{yz} \end{matrix}\right\}_i = [D_s]_i \left\{\begin{matrix} \gamma_{xz} \\ \gamma_{yz} \end{matrix}\right\}_i$$

The potential energies of the upper part and lower part of the laminated plate are

$$\Phi^U = \sum_{i=1}^{N^U} \frac{1}{2} \int (\{\varepsilon\}_i^T [D_e]_i \{\varepsilon\}_i + \{\gamma\}_i^T [D_s]_i \{\gamma\}_i) \, dx \, dy \, dz$$

$$-p \sum_{i=1}^{N^U} \frac{1}{2} \int (\{w'\}^T [N]_i \{w'\}) \, dx \, dy - T^{\{i\}} \tag{4}$$

$$\Phi^L = \sum_{i=1}^{N^L} \frac{1}{2} \int (\{\varepsilon\}_i^T [D_e]_i \{\varepsilon\}_i + \{\gamma\}_i^T [D_s]_i \{\gamma\}_i) \, dx \, dy \, dz$$

$$-p \sum_{i=1}^{N^L} \frac{1}{2} \int (\{w'\}^T [N]_i \{w'\}) \, dx \, dy + T^{\{i\}} \tag{5}$$

where
$$\{w'\} = \begin{Bmatrix} w_x \\ w_y \end{Bmatrix} \qquad\qquad [N]_i = \begin{bmatrix} N_x & N_{xy} \\ N_{xy} & N_y \end{bmatrix}_i$$

p is the scaling parameter of the load and $[N]_i$ is the matrix of resultant forces. Because of the delamination, the upper and the lower plates are only partially joined. $T^{\{i\}}$ is the work done by the internal force between the upper and lower parts, on the upper plate.

Applying the principle of minimum potential energy to the upper plate and the lower plate, respectively, we obtain

$$\delta \Phi^U = 0 \tag{6}$$

$$\delta \Phi^L = 0 \tag{7}$$

3 The FEM and The Asymptotic Method

The analysis is carried out by a combination of the finite element and asymptotic expansion methods. For details, please see References [10] and [11].

4 Results and Discussion

In this section, we study the effects of the shape, position and size of a delamination on the buckling load of a laminated plate. Hereafter, the symbol "//" is used to show the location of the delamination. For example, $(0^\circ//90^\circ/45^\circ)$ means a $(0^\circ/90^\circ/45^\circ)$ plate with a delamination between the 0°-layer and the 90°-layer, $(0^\circ//0^\circ_4)$ means an isotropic plate with a delamination at a distance of H/5 below its top surface. Two different materials of laminates, one isotropic, the other orthotropic, are considered. The material properties are

M-1 $E_1/E_2 = 1$ $G_{12} = G_{13} = G_{23} = E_2/2(1+\nu)$ $\nu_{12} = 0.3$

M-2 $E_1/E_2 = 25$ $G_{12} = G_{13} = 0.25 E_2$ $G_{23} = 0.2E_2$ $\nu_{12} = 0.25$

In all cases, a square plate with L*L dimension is considered. A compressive load p, in the x direction, is applied along the edges $x = \pm L/2$ of the plate. $\varepsilon = H/L = 0.005$.

Table 1		Buckling Load $\lambda = p/E\,\epsilon^2$ (s.s.)		
Mode No.		Square	Circle	Ellipse
M-1 (0°//0°)	1	6.7893	7.0528	6.9149
	2	11.480	11,447	12.442
	3	12.748*	11.530*	15.908*
M-2 (45°//45°)	1	27.028	27.426	28.482
	2	47.195	41.093*	48.451*
	3	48.147*	50.510	58.931

Table 2		Buckling Load $\lambda = p/E\,\epsilon^2$		
delamination shape		Square	Circle	Ellipse
M-1	s.s.	3.5152	3.4848	3.4928
	c.	6.2256	6.2160	6.7328
M-2	s.s.	19.472	19.529	19.689
	c.	30.446	31.770	41.278

Three different delamination shapes, a square, a circle, and a ellipse, of equal area ($\pi L^2/16$) are considered here. In each case, the center of the delamination is coincident with that of the plate.

For the case of symmetric delamination (i.e. where the full plate is symmetric about the delamination), the first buckling mode does not open, while for a plate with an unsymmetric delamination, the first buckling mode opens up, but both the upper and lower parts of the plate deflect in the same direction. For the symmetric case, the opening buckling mode, shown in Figure 2, may occur in the second or third mode. In Table 1, the numbers with *'s are the critical loads for the modes which open up, for a symmetric case. Figure 3 shows the first buckling mode for a (non-symmetric) M-2 material plate (0°/90°//0°/90°/0°) with a square delamination in the center. The delaminated area is again $\pi L^2/16$. Here, as mentioned above, the deflections of the upper and lower parts of the plate are in the same direction. We found this general nature of buckling modes to be true in all our examples.

Table 2 shows the effect of the shape of the delamination on the buckling load. For M-1, the plate is (0^o_2//0^o_3). For M-2, the plate is (0°/90°//0°/90°/0°). From Table 2 we can see that whether the delamination shape significantly affects the buckling load depends on the boundary conditions and the material property. For a square plate, when it is simply supported, the shape of the delamination has negligible effect on the buckling load. However, for a clamped square orthortropic plate, the delamination shape does affect the critical load. As expected, the material has a dramatic effect on the buckling load.

Figure 4 shows the effect of the plate thickness on the dimensionless buckling load λ. ϵ vanishes for a Kirchhoff plate theory. Thus, figure 4 also demonstrates the importance of retaining the shearing strains for moderately thick plates.

In Figure 5, we show the effect of the size of the delamination on the buckling load. In this case, the material is isotropic, i.e. M-1, the delamination shape is square with its center coincident with that of the plate. The plates are (0°//0^o_4). As expected, the buckling load decreases monotonically with the size of the delamination. Small delaminations have very little effect on the buckling load.

The effect of the delamination position on the buckling load is shown in Figure 6. A (0°//0^o_4) simply supported square plate with a square delamination is considered here. The area of the delamination is $L^2/4$. From Figure 6, we can see that the position only moderately affects the buckling load.

The mathematical model used here may be developed further for the study of buckling of laminated plates with more than one delamination region. It also can be used for nonlinear analyses of laminated plates with delaminations.

Acknowledgements

References

[1] Chai,H., Babcock,C.D., Knauss,W.G., " One Dimentional Modelling of Failure in Laminated Plates by Delamination Buckling", Inter. J. Solids and Struc., Vol.17,No.11,(1981), 1069-1083.

[2] Whitcomb,J.D., " Finite Element Analysis of Instability Related Delamination Growth", J. Composite Materials, Vol.15,(1981),403-426.

[3] Evans,A.G., Hutchinson,J.W., " On the Mechanics of Delamination and Spalling in Compressed Films", Inter. J. Solids and Struc., Vol.20,No.5,(1984),455-466.

[4] Yin,W.L., " Buckling and Growth of a Strip Delamination in a General Laminate", The Proceedings of International Symposium on Composite Materials and Structures, 1986,592-597.

[5] Yin,W.L., Wang,J.T.S., " The Energy Release Rate in the Growth of One-Dimentional Delamination", J. Appl. Mech., Vol.51,(1984),939-941.

[6] Yin,W.L., " Axisymmetric Buckling and Growth of a Circular Delamination in a Compressed Laminate", Inter. J. Solids and Struc., Vol.21,(1985),503-514.

[7] Fei,Z.Z., " Analysis of Stability for the Delamination of Clamped Orthotropic Rectangular Plates", The Proceedings of International Symposium on Composite Materials and Structures, 1986,207-213.

[8] Seide,P., " An Improved Approximate Theory for the Bending of Laminated Plates", Mechanics Today, Pergammon, Oxford, Vol.5,(1980), 451-466.

[9] Xie,Y., Xie,Z., " A new Variational Approach for Mindlin Plate", The Proceedings of International Conference on Computational Engineering Mechanics, Atlanta,USA, Apr.1988.

[10]Mukherjee,Y.X., Xie,Z., Ingraffea,A., " Delamination Buckling of Laminated Plates", Submitted for publication.

[11]Xie,Y., " A Study on Stresses and Delamination Buckling of Laminates", Ph.D. theses, Tsinghua University, PR CHINA, in chinese.
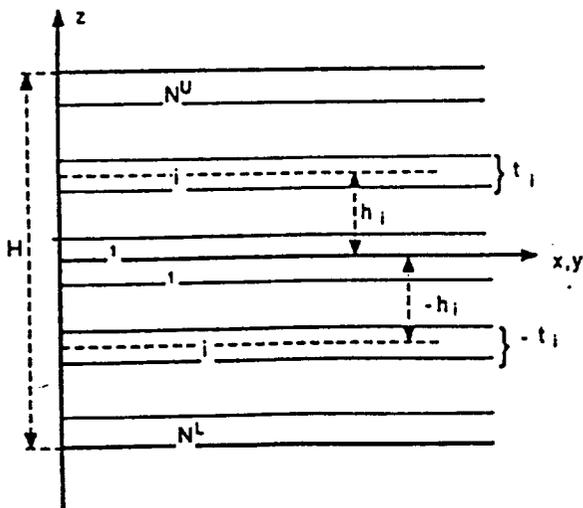
Figure 1   Coordinate system and geometrical parameters
            of the laminated plate
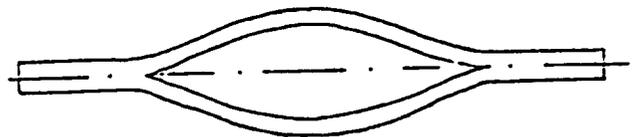
Figure 2

Cross-section of the lowest opening buckling mode
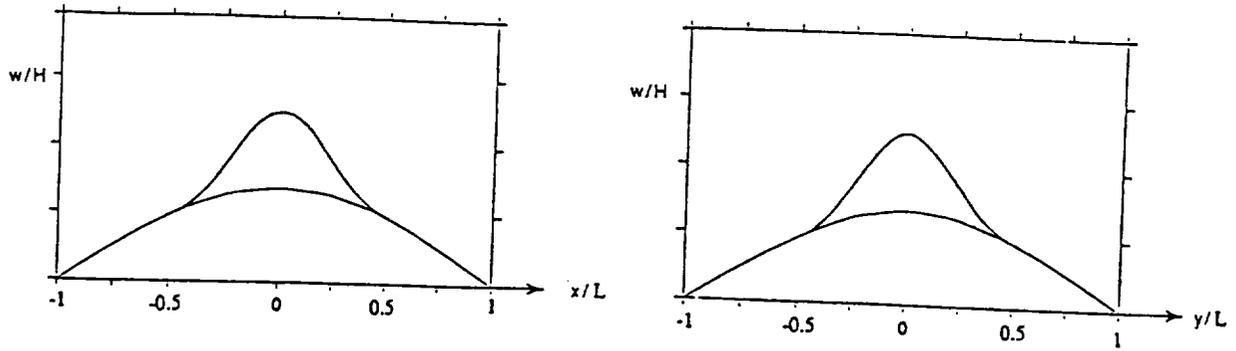
Figure 3   Cross-sections of the buckling mode of the delaminated plate.
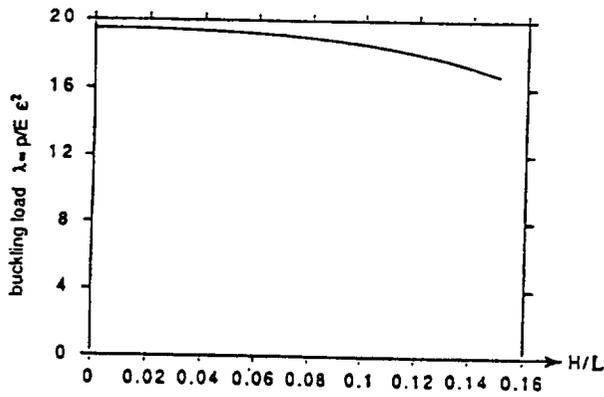
Figure 4   Effect of the plate thickness on the buckling load for a square delamination
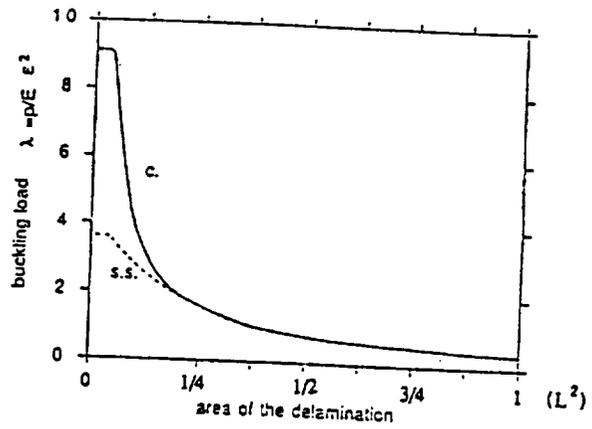
Figure 5.   Variations of the buckling loads as functions of the area of a (square) delamination, for two boundary conditions
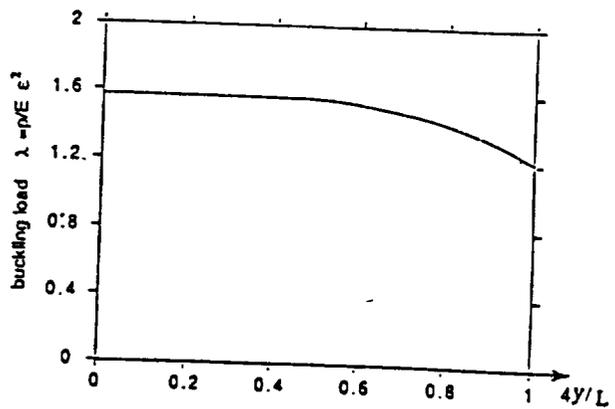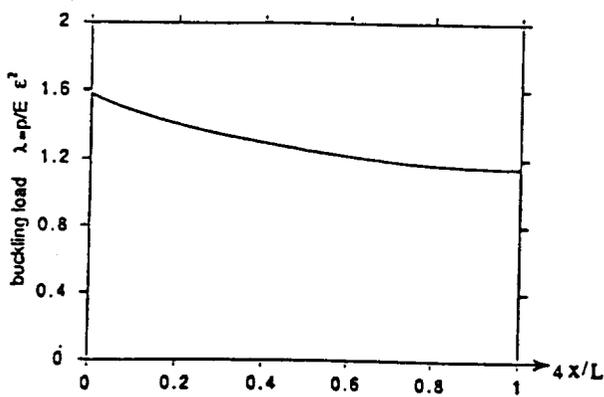
Figure 6   Variations of the buckling load as functions of the position of the delamination. The center of the delamination is at (x,0) and (y,0),respectively

487

BENDING AND STRETCHING FINITE ELEMENT ANALYSIS OF ANISOTROPIC VISCOELASTIC COMPOSITE PLATES [1]

Harry H. Hilton[2] and Sung Yi[3]
University of Illinois at Urbana-Champaign

N94- 71485

ABSTRACT

Finite element algorithms have been developed to analyse linear anisotropic viscoelastic plates, with or without holes, subjected to mechanical (bending, tension), temperature and hygrothermal loadings. The analysis is based on Laplace transforms rather than direct time integrations in order to improve the accuracy of results and save extensive computational time and storage. The time dependent displacement fields in the transverse direction for the cross ply and angle ply laminates are calculated and the stacking sequence effects of the laminates are discussed in detail. Creep responses for the plates with or without a circular hole are also studied. The numerical results compare favorably with analytical solutions, i.e. within 1.8% for bending and 10^{-3}% for tension. The tension results of the present method are compared with those using the direct time integration scheme.

1. INTRODUCTION

Advanced composite laminates are used in atmospheric and space flight vehicles in order to improve performance by substantial structural weight savings. The polymer matrix exhibits degradation of material mechanical properties when exposed to hygrothermal environment and the hygrothermal expansion induces residual stresses in a composite laminate which may result in delamination and subsequent structural failures [1, 2]. Therefore, time dependent analyses of polymer composite structures are required to predict structural lifetimes under short and long term loading, mandating anisotropic viscoelastic analyses. The linear thermo-viscoelastic theory for anisotropic nonhomogeneous materials whose stress-strain relations are expressed by hereditary integrals has been formulated by Hilton and Dong [3]. Finite element methods (FEM) are the most powerful tool for the analysis of complicated systems. However solving time dependent problems using FEM is difficult and requires enormous computational time and memory storage, since the constitutive law for linear viscoelastic materials is described by hereditary integrals or fractional differential operators as developed by Rogers [4]. Numerical procedures for the analysis of viscoelastic boundary value problems have been proposed [5-8]. Taylor *et al.* [6] developed a finite element procedure to analyse isotropic linear viscoelastic solids undergoing mechanical and thermal deformations. The integration of the governing equations is performed step by step using a finite difference recurrence relationship for approximate calculations of displacement derivatives. This method requires storage of only the previous time solution instead of all the solutions throughout the loading time history. Srinatha *et al.* [7] suggested a similar numerical procedure for an isotropic material and applied it to solve plane problems using the trapezoidal integration method previously developed by Zak [5] to evaluate integral equations. However, the accuracy of the direct integration scheme primarily depends upon the size of the time step Δt. Moreover since the solutions of rate dependent problems at the present time are affected by the previous solutions, numerical errors may be accumulated throughout the time history and such error accumulations are described in [6,8]. If the value of time step is decreased in order to obtain better approximations, the number of iterations will be augmented and the computational time will be significantly increased. Use of the direct integration method for long-term predictions of time dependent dimensional changes and stresses in composite structures may require huge amounts of computational time.

In the present study, a numerical algorithm is developed for the efficient analysis of time dependent deformations and stresses in linear viscoelastic composite materials which are subjected to mechanical, temperature and hygrothermal loadings. For time dependent temperatures and/or moisture contents, the hereditary time integrals are not convolution ones and approximate elastic-viscoelastic analogies [9,10] have been developed allowing the use of Laplace or Fourier transforms. In the present analysis the FEM equations are formulated in real time and then converted to reduced times, thereby not requiring the above approximate formulations. The Laplace transform technique is adopted to improve the accuracy of solutions, to save expensive computational time and to reduce the tedious formulation of numerical procedures.

---

[2] Professor of Aeronautical and Astronautical Engineering
[3] Research Assistant, AAE Department

2. ANALYTICAL FORMULATION

2.1 Governing Equations for Linear Thermo-Viscoelastic Composite Plates

In the principal material coordinates x_i, the constitutive relationship for anisotropic linear viscoelastic materials can be described by the following hereditary integrals [3]

$$\sigma_{ij}(T, M, \mathbf{x}, t) = \int_{-\infty}^{t} C_{ijkl}(T, M, t, \tau) \frac{\partial}{\partial \tau}[\epsilon_{kl}(\mathbf{x}, \tau) - \epsilon_{kl}^{*}(\mathbf{x}, \tau)] \, d\tau \tag{1}$$

where σ_{ij} are stress components at time t and ϵ_{kl} and ϵ_{kl}^{*} are total strains and free hygrothermal strains respectively. The coefficients of hygrothermal expansions are assumed to be independent of time, temperature and moisture. The free hygrothermal strains ϵ_{ij}^{*} may then be expressed as

$$\epsilon_{ij}^{*} = \alpha_{ij}\Delta T + \beta_{ij}\Delta M \tag{2}$$

where α_{ij} and β_{ij} are respectively thermal and hygroscopic expansion coefficients and $\Delta T(\mathbf{x}, t)$ and $\Delta M(\mathbf{x}, t)$ are the temperature and moisture changes related to an unstressed reference state. For thermorheologically simple materials [3,12], the relaxation moduli can be represented in the form

$$C_{ijkl}(T, M, t) = C_{ijkl}[T_r, M_r, \varsigma_{ijkl}(\mathbf{x}, t)] \tag{3}$$

where the subscript r denotes reference conditions and the ς_{ijkl} are reduced times, which are related to the shift functions in the following manner

$$\varsigma_{ijkl}(\mathbf{x}, t) = \int_{0}^{t} a_{ijkl}(T(\mathbf{x}, \tau), M(\mathbf{x}, \tau)) \, d\tau \tag{4}$$

The introduction of these reduced times changes Eqs. (1) to convolution ones in the ς plane, however x derivatives in the field equations take on new definitions, i.e.

$$\left(\frac{\partial}{\partial x_j}\right)_t = \left(\frac{\partial}{\partial x_j}\right)_\varsigma + \left(\frac{\partial \varsigma}{\partial x_j}\right)_t \left(\frac{\partial}{\partial \varsigma}\right)_x \tag{5}$$

While the ς derivatives can be computed, their accuracy is low since the shift functions in Eq. (4) are experimentally determined. The present method avoids such pitfalls by determining all derivatives in the x plane and then at later stages transforming the equations into the ς plane. Composite viscoelastic material properties can be characterised by master relaxation curves determined from tensile relaxation experiments [13-16]. The modulus C_{1111} is assumed time independent since the x_1-direction is dominated by the fibers. All ς_{ijkl} in Eq. (4) are assumed identical and equal to ς. This allows one to express the relaxation moduli in terms of Prony series [11] such that

$$C_{ijkl}(\varsigma) = C_{ijkl}^{0} + \sum_{\rho=1}^{N} C_{ijkl}^{\rho} \, exp(-\varsigma/\lambda_\rho) \tag{6}$$

where the constants λ_ρ are relaxation times, N is the number of terms for the series expansion and C_{ijkl}^{0} and C_{ijkl}^{ρ} must be time independent and symmetric. The stress-strain relation (1) now reduces to [12]

$$\hat{\sigma}_{ij}(\mathbf{X}, \varsigma) = \int_{-\infty}^{\varsigma} \hat{C}_{ijkl}[T_r, M_r, \varsigma - \varsigma'] \frac{\partial}{\partial \varsigma'}[\hat{\epsilon}_{kl}(\mathbf{X}, \varsigma') - \hat{\epsilon}_{kl}^{*}(\mathbf{X}, \varsigma')] \, d\varsigma' \tag{7}$$

with $\mathbf{X}$ are the laminate coordinates and $\hat{C}_{ijkl}$ are the moduli in $\mathbf{X} - \varsigma$ plane.

2.2 A Variational Formulation for Thermo-Viscoelastic Problems

The variational functional for linear thermo-viscoelastic problems has been given by Gurtin [17] and in the absence of body forces, its first variation is

$$\delta \pi = \int_{V} \int_{s=-\infty}^{s=t} \int_{\tau=-\infty}^{\tau=t-s} \bar{C}_{ijkl}(T, M, t, s, \tau) \frac{\partial}{\partial \tau} \{\bar{\epsilon}_{ij}(\mathbf{X}, \tau) - \bar{\epsilon}_{ij}^{*}(\mathbf{X}, \tau)\} \, d\tau$$

489

$$\cdot \frac{\partial \delta \bar{\varepsilon}_{kl}(\mathbf{X}, s)}{\partial s} ds\ dV - \int_{S_T} \int_{s=-\infty}^{s=t} \Omega_i(t-s) \frac{\partial \delta u_i(\mathbf{X}, s)}{\partial s} ds\ dS_T = 0 \tag{8}$$

where V is the volume of viscoelastic solid, S_T is the surface on which tractions Ω_i are applied and u_i are the displacements. This relation forms the basis of finite element linear viscoelastic boundary value problem formulations.

2.3 Finite Element Formulation Using Integral Transforms

Using the aforementioned variation principles, the finite element equilibrium equations for linear viscoelastic solids are obtained in the real time domain. Then substitution of Eq. (4) and (6) into those equilibrium equations and taking the Laplace transform result in a system of algebraic equations

$$[\ K_{mn}^0 + K_{mn}^t s \tilde{f}(s)\ ] \tilde{U}_n(s) = \tilde{F}_m(s) + \tilde{F}_m^{th}(s) \tag{9}$$

In the above, K_{mn}^0 and K_{mn}^t are the global stiffness matrices, U_n is the global vector of nodal displacements, $\tilde{F}_m$ and $\tilde{F}_m^{th}$ are L.T. of global nodal force vectors due to applied tractions and hygrothermal gradients respectively and $\tilde{f}(s)$ is the L.T. of the exponential functions in Eq. (6). It should be noted that Eq. (9) is identical to the equivalent elastic problem except for the equivalent viscoelastic moduli [12]. One approach to solving the above system is to carry out an orthogonal transformation which simultaneously diagonalizes the two real symmetric matrices: K_{mn}^0 and K_{mn}^t. Once the nodal displacements are determined from (10) in the Laplace space, they can be transformed back into real time at selected nodes where critical conditions of interest occur thus saving additional computational time. If the loading functions are provided in analytic forms, the partial-fraction method produces excellent results. Even if those forces are given numerically, diverse Laplace transform and numerical L.T. inversion techniques are available [18-21]. In Ref. 21, eight numerical algorithms of the Laplace transform inversion methods are compared against each other. The study shows that Schapery's collocation method [18] and Becker's multidata method [19] provide good results for non-oscillatory functions of time with less computer time and that Durbin's inversion method [20] based on fast Fourier sine-cosine transformations yields accurate results for the oscillatory time functions, but at the expense of much computing time.

3. NUMERICAL RESULTS

3.1 Anisotropic Plate Bending Studies

Consider a thin composite laminate with symmetry in both geometry and material properties about the middle plane and subjected to bending moments and lateral loading and assume the Kirchhoff hypotheses. Two studies were conducted to verify the viscoelastic finite element program VBEND [10] which are developed to analyze viscoelastic bending responses of composite plates. The first study concentrates on elastic results and the second focuses on time dependent solutions. The dimensions of the composite laminate are 100 in x 100 in and the thickness of the plate is 1 in. A total of 128 elements and 243 degrees of freedom are used for these studies. At time t=0, the viscoelastic finite element solutions are compared with the corresponding elastic results presented in Ref. 22 for composite laminates with fully clamped edges. The elastic orthotropic material properties of the composite lamina are $E_{11} = 10^7$ psi, $E_{22} = 10^6$ psi, $\nu_{12} = 0.3$, and $G_{12} = 0.25 \times 10^6$ psi [22]. The laminate is subjected to uniformly distributed loading p = 0.02 psi. The maximum deflections at the center of the laminates are compared with those of [22] for various orientation of the principal orthotropic material axes with respect to the laminate axes. As shown in Fig. 1, at time t=0, the viscoelastic finite element solutions agree very well within 3% with the elastic solutions of [22]. In the second study, viscoelastic finite element solutions calculated by VBEND [10] are compared with analytic results evaluated by the viscoelastic-elastic analogy [3]. Time dependent deflections for a simply-supported composite plate subjected to the same load as the first example are considered. The elastic anisotropic bending stiffnesses of the laminate are $D_{2222}/D_{1111} = 1, (D_{1122} + 2D_{1212})/D_{1111} = 1.5$, and $D_{1112}/D_{1111} = D_{2212}/D_{1111} = -.5$. Using the viscoelastic-elastic analogy, the maximum time dependent deflections for the anisotropic plate with the above flexural stiffnesses, simply supported along all the edges and subjected to a uniformly distributed loading $p(t)$ can be expressed in the Laplace domain as [3,20]

$$\tilde{W}_{max}(s) = \frac{0.00452\ a^4\ \tilde{p}(s)}{\tilde{D}_{1111}(s)} \tag{10}$$

where $\sim$ denote the Laplace transform. Temperature shift factor is equal to one and the time function for all the flexural stiffnesses D_{ijkl} is taken as

$$f(t) = 6.698 \times 10^{-2} + 0.93302 \cdot exp(-t/1000) \tag{11}$$

490

Maximum deflections computed by VBEND [10] and the analytic ones calculated by Eq.(11) are plotted in Fig. 2. Excellent agreement between the two solutions is obtained within a 1.8% maximum error.

Consider next simply supported square composite plates which are subjected to a uniformly distributed time-independent loading p=0.03 psi. The time dependent displacement fields in the transverse direction for various symmetric cross and angle ply laminates such as $(0_5/90_5)_s$, $(0/90)_{5s}$, $(45_5/-45_5)_s$, $(45/-45)_{5s}$, $(30_5/-30_5)_s$, and $(30/-30)_{5s}$ are calculated. In these studies, the elastic orthotropic material properties and the dimensions of the lamina are also the same as those used for the first verification study. The thickness of a lamina is 0.05 in and 20 plies were used. The time varying function for the relaxation moduli is defined by Eq.(11) and C_{1111} is assumed as time independent since E_{11} and ν_{12} are not sensitive to time as stated earlier. The maximum deflections are observed at the center of plates for all the cases. The maximum displacements for the $(0_5/90_5)_s$ and $(0/90)_{5s}$ laminates are displayed in Fig. 3 as functions of time. At time t=0, the maximum deflection calculated for both the $(0_5/90_5)_s$ and $(0/90)_{5s}$ laminates is 0.04546 in. The maximum deflections for those laminates increase about 23% after 2×10^4 second, while the relaxation moduli such as $C_{1122}(t)$, $C_{2222}(t)$, and $C_{1212}(t)$ degrade 93.3% for the same time period. The numerical results show that the maximum deflections for symmetric cross ply laminates under the lateral uniform loading are not affected by the stacking sequence. However, the stacking sequence of the laminate has significant influence on the solutions obtained for the symmetric angle ply laminates. Comparisons of viscoelastic maximum deflections of the $(45_5/-45_5)_s$ laminate with the $(45/-45)_{5s}$ laminate indicate that the maximum transverse displacements of the $(45_5/-45_5)_s$ laminate are larger than those of the $(45/-45)_{5s}$, shown in Fig. 4. At time t=0, the values calculated for the maximum displacements in X_3 direction of the $(45_5/-45_5)_s$ and $(45/-45)_{5s}$ laminates are 0.03212 in and 0.02778 in, respectively. These results can be readily explained if we note that the flexural stiffnesses D_{1112} and D_{2212} for the $(45_5/-45_5)_s$ laminate are smaller than those for the $(45/-45)_{5s}$ laminate and the exact elastic maximum deflections obtained from Reference 20 including the D_{1122} and D_{1112} terms are larger than specially orthotropic solution calculated by ignoring those twist coupling terms. The maximum deflection of the $(45_5/-45_5)_s$ laminates increases 24% for the 2×10^4 second period while that of the $(45/-45)_{5s}$ laminate increases only 11%. The solutions for the $(30_5/-30_5)_s$ and $(30/-30)_{5s}$ laminates are depicted in Fig. 5 which shows that the rates of change of the maximum deflections of the $(30_5/-30_5)_s$ and $(30/-30)_{5s}$ laminates increase by 26% and 15% respectively.

3.2 Plane Stress Anisotropic Plate Studies

In these studies, creeps of unnotched graphite/epoxy composite plates are calculated. The relaxation moduli and shift factors are the same as those used in Ref. 8. The solutions evaluated by Ref. 8 using the direct integration method and the present results are compared with the analytic ones obtained from the viscoelastic-elastic analogy for the classical lamination theory. At the 75° F reference temperature, a time independent stress of 1,728 psi is applied at t = 0 and is unloaded instantaneously after 24 hours. Creep and creep recovery responses in the $(45/-45)_s$ laminate are given in Table 1. At t=0, the three results are identical. However, after 24 hours, the error between analytic and Ref. 8 results is 5%. Another comparison study between the transformation method and the direct integration method is conducted using the various time-step sizes ($\Delta t = 200$ and 1000 sec.). The results for these cases and those calculated by the present transformation method and by the elastic-viscoelastic analogy are depicted in Fig. 6. Excellent agreement is obtained between creep results evaluated using the present method and the analytic one.

Table 1. Comparison for Creep Strains

Time (hrs)	Ref. 8	Present	Analytic Solution
0	0.564×10^{-3}	0.564×10^{-3}	0.564×10^{-3}
24	0.625×10^{-3}	0.605×10^{-3}	0.605×10^{-3}
48	0.607×10^{-4}	0.636×10^{-4}	0.636×10^{-4}

Creep responses in laminates subjected to quasi-static sinusoidal loadings are also studied. The maximum amplitude of loading is 10^{-3} psi and the operating temperature is taken as 122° F and 160° F. The maximum strain amplitudes in the $(45/-45)_s$ laminates are given in Fig. 7 with respect to the driving frequencies. At the same frequency, the maximum strain amplitudes at T = 160° F are larger than those at T = 122° F and reach equilibrium at about $\omega = 10^{-13}$. Temperature and frequency shift behaviors in the viscoelastic composites can also be observed in these resluts.

Creep responses for composite plates with a circular hole are evaluated for viscoelastic GY70/339 composite material properties given in [15] and for laminations of $(0/90)_s$ and $(45/-45)_s$. The dimensions of a plate is 2 in x 4 in and the diameter of the hole is .25 in. 650 degrees of freedom were used. The finite element mesh pattern is depicted in Fig. 8 and for operating temperatures are 75° F and 122° F, a uniaxial tensile stress of $\sigma_x = 5$ psi was uniformly applied along the edge of the plate. The circumferential creep strains around the hole in $(0/90)_s$ and $(45/-45)_s$ laminates are plotted in Fig. 9 and Fig. 10. Maximum circumferential creep strains occur at the 90° angular position in both plates. At $t = 0$, the strains at $\varphi = 90°$ are 1.725×10^{-6} for $(0/90)_s$ and 4.034×10^{-6} for $(45/-45)_s$. Creep strains for $(0/90)_s$ laminate increase about 10.6 % at 75° F and 19.7% at 122° F for the 1 year period, while those in $(45/-45)_s$ plate increase 22.8% at 75° F and 44.2% at 122° F for the same time period. At $\varphi = 42°$ the strains in $(45/-45)_s$ laminate and at $\varphi = 0°$ for the $(0/90)_s$ laminate remain time independent.

4. CONCLUSIONS

Viscoelastic bending and stretching responses of polymer matrix composites have been evaluated using the finite element method and Laplace transform technique. Verification studies show that the L.T. results agree well with the analytical ones. Plate bending time dependent displacement fields in the transverse direction for simply supported square composite laminates have been computed. It is observed that the stacking sequence of the laminates significantly affects the time dependent displacement field for symmetric angle ply laminates. Maximum strains are obtained at $\varphi = 90°$ for both GY70/339 $(0/90)_s$ and $(45/-45)_s$ laminates. The rate of change of creep in $(0/90)_s$ laminate is smaller than that in $(45/-45)_s$ laminate for tensile loads. The present method can be readily applied to any viscoelastic boundary value problems with complex geometries. The advantages of this algorithm as compared with the direct integration method is accuracy, saving of large amounts of computational time for the analysis of long time behavior and the significant reduction of labor for numerical procedures and programming.

REFERENCES

1. Flaggs, D.L., and Crossman, F.W., "Analysis of the Viscoelastic Response of Composite Laminates During Hygrothermal Exposure," J. Composite Materials, Vol. 15, 1981, p. 21.

2. Gibbins, M.N., "Environmental Exposure Effects on Composite Materials for Commercial Aircraft," NASA-15148, 11th Quarterly Progress Reports, June 1981.

3. Hilton, Harry H. and Dong, Stanley B., "An Analogy for Anisotropic, Nonhomogeneous, Linear Viscoelasticity Including Thermal Stresses," Proc. of 8th Midwestern Mechanics Conference, 1964, p. 58.

4. Rogers, Lynn, "Operators and Fractional Derivatives for Viscoelastic Constitutive Equations," J. of Rheology, Vol. 27, 1983, p. 351.

5. Zak, A.R., "Structural Analysis of Realistic Solid Propellant Materials," J. Space Craft and Rockets, Vol. 5, 1967, p. 270.

6. Taylor, R.L., Pister, K. S., and Goudreau, G.L., "Thermomechanical Analysis of Viscoelastic Solids," International Journal for Numerical Methods in Engineering, Vol. 2, 1970, p. 45.

7. Srinatha, H.R. and Lewis, R.W., "A Finite Element Method for Thermoviscoelastic Analysis of Plane Problems," Comp. Meth. Appl. Mech. Eng., Vol. 25, 1981, p. 21

8. Lin, K.Y. and Hwang,I.H., "Thermo-Viscoelastic Analysis of Composite Materials," Journal of Composite Materials, Vol. 23, 1989, p. 554.

9. Hilton, Harry H. and Russell, H.G., "An Extension of Alfrey's Analogy to Thermal Stress Problems in Temperature-Dependent Linear Viscoelastic Media," J. of Mechanics and Physics of Solids, Vol. 9, 1961, p. 152.

10. Hilton, Harry H. and Clements, J.R., "Formulation and Evaluation of Approximate Analogies for Temperature-Dependent Linear Viscoelastic Media," Proc. Conf. on Thermal Loading and Creep, Inst. of Mech. Eng., London, 1964, p. 6-17.

11. Hilton, Harry H. and Yi, Sung, "Finite Element Analysis of Viscoelastic Composite Materials," University of Illinois U-C, Technical Report UILU ENG 89-0507, December 1989.

12. Hilton, Harry H., "Viscoelastic Analysis," Engineering Design for Plastics, Reinhold Pub. Corp., New York, 1964, p. 199.

13. Jones, R.M., Mechanics of Composite Materials, Scripta Boo Company, Washington, D.C., 1975.

492

14. Kibler, K.G., "Effects of Temperature and Moisture on the Creep Compliance of Graphite-Epoxy Composites," Effect of Service Environment on Composite Materials, AGARD - CP - 288, Rue Anuelle, 92200 Neuilly, Sur Seine, France.

15. Crossman, F.W., Mauri, R.E., and Warren, W.J., "Moisture Altered Viscoelastic Response of Graphite/Epoxy Composite," Advanced Composite Materials-Environmental Effects, ASTM STP 658, J.R. Vinson, Ed., American Society for Testing and Materials, 1978, p. 205.

16. Tuttle, M.E., and Brinson, H.F., "Accelerated Viscoelastic Characterization of T300/ 5208 Graphite-Epoxy Laminates," NASA Contractor Report 3871, 1985.

17. Gurtin, M.E., "Variational Principles in the Linear Theory of Viscoelasticity," Archive Rat. Mech. and Analysis, Vol. 13, 1963, p. 179.

18. Schapery, R.A., "Approximate Methods of Transform Inversion for Viscoelastic Stress Analysis," Proc. 4th U.S. Natl. Congr. Appl. Mech., ASME, 1962, p. 1075.

19. Cost, T.L. and Becker, E.B., "A Multidata Method of Approximate Laplace Transform Inversion," International Journal for Numerical Methods in Engineering, Vol. 2, 1970, p. 207.

20. Durbin, F., "Numerical Inversion of Laplace Transforms: An Efficient Improvement to Dubner and Abate's Method," Comp. J., Vol. 17, 1974, p. 371.

21. Narayanan, G. V. and Beskos, D. E., "Numerical Operational Methods for Time-Dependent Linear Problems," International Journal for Numerical Methods in Engineering, Vol. 18, 1982, p. 1829.

22. Ashton, J.E. and Whitney, J.M., Theory of Laminated Plates, Technomic Publishing Co., Inc., 1970.
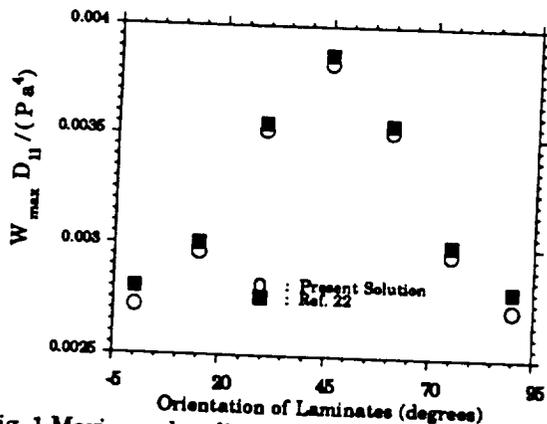
Fig. 1 Maximum bending deflection of square clamped composite laminate.
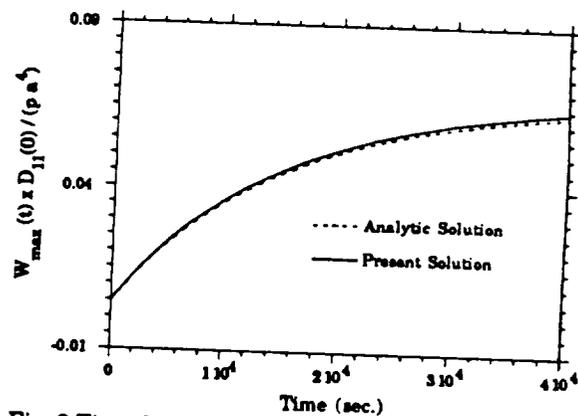
Fig. 2 Time dependent max. bending deflection of a simply supported composite plate.
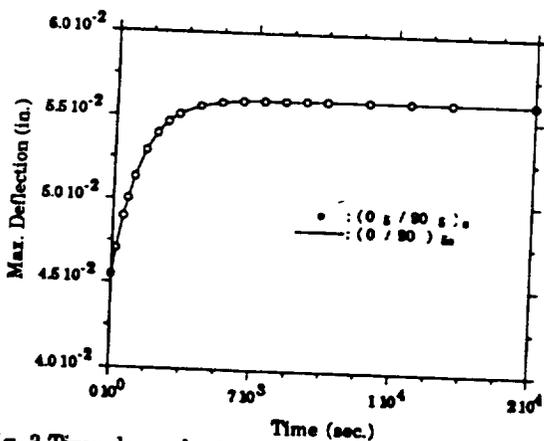
Fig. 3 Time dependent max. bending deflections of simply supported $(0_5/90_5)_s$ and $(0/90)_{5s}$ laminate.
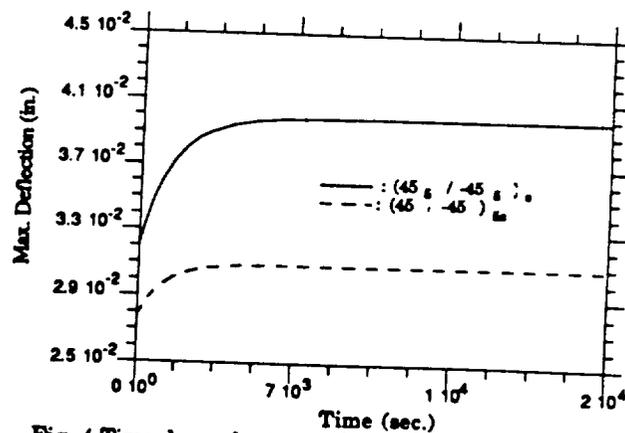
Fig. 4 Time dependent max. bending deflections of simply supported $(45/-45)_{5s}$ and $(45_5/-45_5)_s$ laminate.
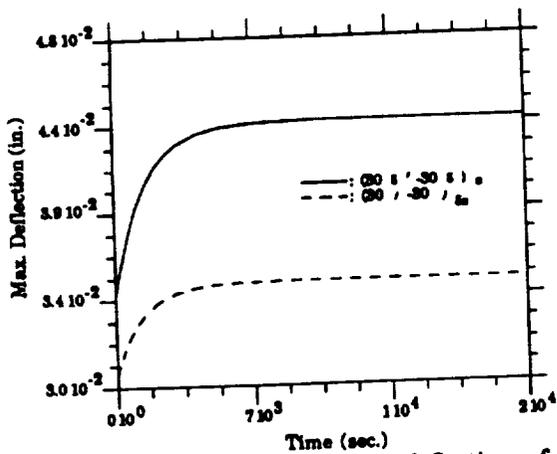
Fig. 5 Time dependent max. bending deflections of simply supported $(30/-30)_{5s}$ and $(30_5/-30_5)_s$ laminate.
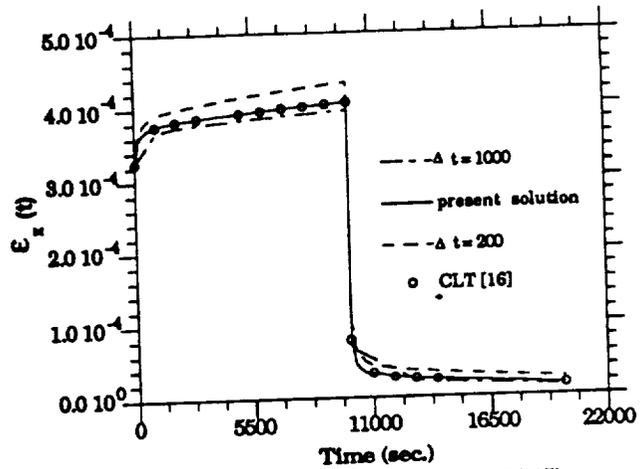
Fig. 6 Time dependent in-plane strain of $(45/-45)s$ graphite/epoxy laminate (Temp.=140 °F).
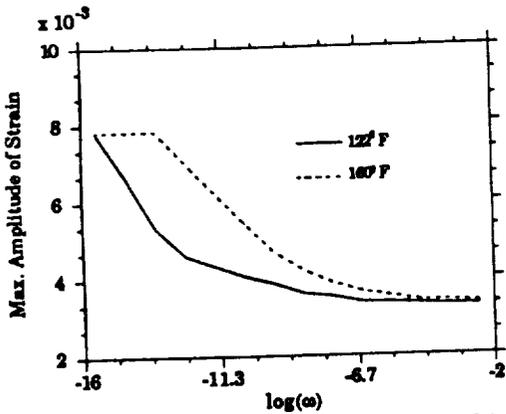
Fig. 7 Max. strain amplitudes of $(45/-45)_s$ graphite/epoxy laminate vs frequency.
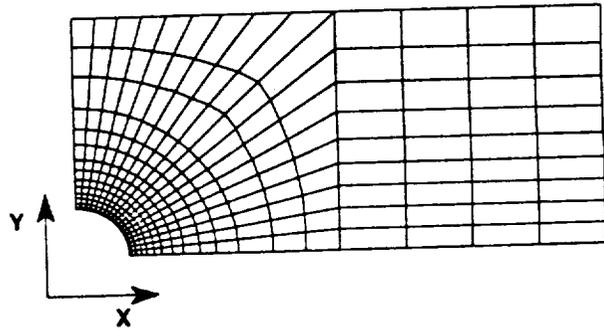
Fig. 8 Finite elment mesh for GY70/339 composite plate with a circular hole
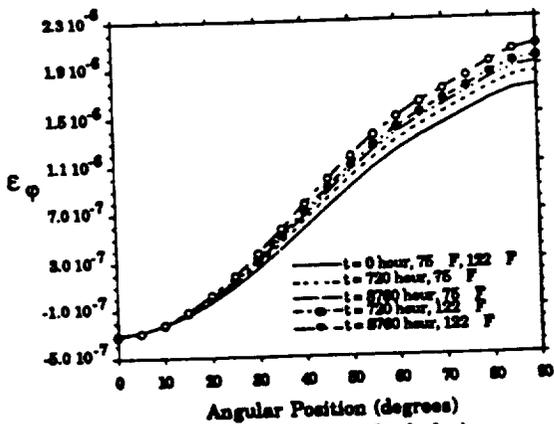
Fig. 9 Circumferential creep around a hole in GY70/339 $(0/90)_s$ laminate
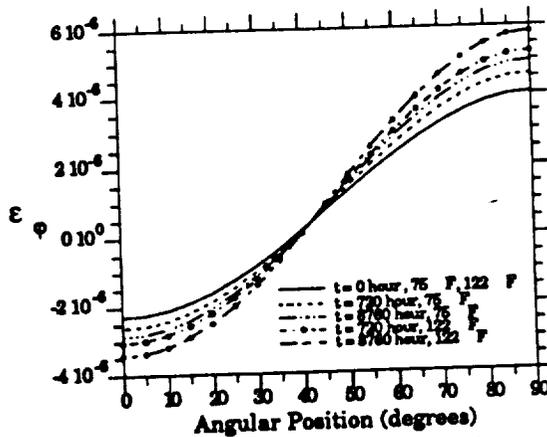
Fig. 10 Circumferential creep around a hole in GY70/339 $(45/-45)s$ laminate

OPTIMIZATION OF STRUCTURE-CONTROL SYSTEMS WITH EFFICIENCY CONSTRAINT

N94- 71486

H. Öz
Dpartment of Aeronautical and Astronomical Engineering
The Ohio State University, Columbus, Ohio 43210

N. S. Khot
Flight Dynamics Laboratory (WRDC/FIBRA)
Wright–Patterson Air Force Base, Ohio 45433-6553

Abstract

The structure-control system optimization problem is formulated with constraints on the closed–loop eigenvalues and the efficiency of the reduced order system. The feasibililty of the approach is illustrated by designing the ACOSS-FOUR structure with a reduced order system and improving the efficiency characteristics of the structures–control system.

1. Introduction

The efficiency of a structure-control system is a nondimensional parameter which indicates the fraction of the total control power expended usefully in controlling a finite-dimensional system. The balance of control power is wasted on the truncated dynamics serving no useful purpose towards the control objectives. In References 1 and 2 it has been demonstrated that the concept of efficiency can be used to address a number of control issues encountered in the control of dynamic systems such as the spillover effects, selection of a good controller configuration and obtaining reduced order control models. An important aspect of the efficiency approach to structure-control system is that the behavior of the full-order system can be ascertained based on the reduced-order design model without any knowledge of the truncated system dynamics. The efficiency compares the control power lost to the truncated dynamics thereby not serving the purpose of control, to the total control power expended via the reduced-order control design model on the full-order physical system.

The subject of structure-control system optimization is inherently multidisciplinary (Refs. 3 and 4). A variety of objectives and constraints can be proposed both at the system-level and subsystem-levels (structure or control subsystems) to bring about an interdisciplinary study of the problem. For obvious reasons, an ultimate objective for space-structures is to have a minimum mass structure subject to structural and/or control system constraints.

A key feature in controlling a reduced-order model of a high dimensional structural system must be to achieve high power efficiency of the control system while satisfying the control objectives and constraints. As a further enhancement of the optimization formulation presented in reference 4. this paper brings on the power efficiency of the system into the picture.

2. Efficiency Modes of a Structure-Control System

Consider an Nth order FEM evaluation model of the structural system

$$\mathbf{M}\ddot{q} + \mathbf{E}\dot{q} + \mathbf{K}q = \mathbf{D}f(t) \tag{1}$$

where M, K and D are the mass, stiffness and input influence matrices. $q(t)$ is the vector of nodal displacements. To control the structure described by (1), reduced-order modal state-space equations are considered

$$\dot{x} = \mathbf{A}x + \mathbf{B}f(t)$$

$$x = [\, \eta_c^T \quad \dot{\eta}_c \,]^T \tag{2}$$

where η_c are the $n < N$ structural modes controlled. Hence, considering the structural model problem associated with (1) and denoting the orthonormalized modal matrix Φ of the full order evaluation model we have

$$q = \Phi\eta = [\Phi_c \quad \Phi_R] \begin{bmatrix} \eta_c \\ \eta_R \end{bmatrix} \qquad (3)$$

where R denotes truncated structural modes. The modal-state space system of (2) is the reduced $2n^{th}$ order control design model. The $\mathbf{A}$ and $\mathbf{B}$ matrices have the form

$$\mathbf{A} = \left[\begin{array}{c|c} 0 & I \\ \hline -\omega^2 & -2\varsigma\omega \end{array} \right] \qquad (4)$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ \Phi_c^T \mathbf{D} \end{bmatrix} \qquad (5)$$

where $\omega^2 = \text{diag}\,[\omega_1....\omega_n]$ with ω_r a natural frequency and I is the nth order identity matrix.

Due to any arbitrary input $f(t)$ the control power associated with the input on the actual full-order evaluation system (1) is given by the integral

$$S^R = \int \underset{\sim}{f}^T \mathbf{D}^T \mathbf{M}^{-1} \mathbf{D} \underset{\sim}{f}\, dt \qquad (6a)$$

The portion of this total expended power on the actual physical system that is projected onto reduced-order dynamic system represented by (2) is

$$S_c^m = \int \underset{\sim}{f}^T \mathbf{B}^T \mathbf{B} \underset{\sim}{f}\, dt \qquad (6b)$$

We refer to S^R as the real (total) control power expended and S_C^M as the modal control power expended on the modal control design model. One has (Ref. 1)

$$S^R \geq S_C^M \qquad (7)$$

and the control power wasted to the truncated dynamics is

$$S_u^m = S^R - S_c^m \qquad (8)$$

The model input power efficiency is defined as

$$e\% = \frac{S_c^m}{S^R} \times 100 \qquad (9)$$

with a maximum possible efficiency of 100%.

Associated with e, a power quotient can be defined as

$$Sq\% = \frac{S_u^m}{S^R} \times 100 = (1 - e) \times 100 \qquad (10)$$

We note that while S_C^M is indicative of a quantity for the reduced control design model through the appearance of the B matrix. S^R is a quantity for the evaluation model through the appearance of the mass matrix M. This observation establishes that the model efficiency relates the power performance of the full-order evaluation model of the actual physical system. Most importantly the definition of model efficiency is valid regardless of the specific functional dependence of the input field $f(t)$ which is the physical input to the real system. For example, it does not matter from the point of definition whether $f(t)$ is a control input or not.

Specifically, however, if the input $f(t)$ on the physical system has the functional form of the state-feedback of the reduced-control design model (2) as:

$$f = -Gx$$

(11)

where G is a stabilizing constant control feedback gain matrix of dimension $m \times 2n$, then it can be shown that S_C^M, S^R become:

$$S^R = x_o^T \mathbf{P}^R x_o, \qquad S_c^m = x_o^T \mathbf{P}_c^m x_o, \qquad x_o = x(t_o)$$

(12)

P^R and P_C^M are symmetric positive definite matrices referred to as real and modal control power matrices, respectively. They are the solutions of the Lyapunov equations associated with the closed-loop control system

$$\mathbf{A}_{cl}^T \mathbf{P}^R + \mathbf{P}^R \mathbf{A}_{cl} + \mathbf{G}\mathbf{D}^T \mathbf{M}^{-1} \mathbf{D}\mathbf{G} = 0$$

(13)

$$\mathbf{A}_{cl}^T \mathbf{P}_c^m + \mathbf{P}_c^m \mathbf{A}_{cl} + \mathbf{G}\mathbf{B}^T \mathbf{B}\mathbf{G} = 0$$

(14)

$$\mathbf{A}_{cl} = \mathbf{A} + \mathbf{B}\mathbf{G}$$

(15)

Both power matrices are $2n^{th}$ order, they are computed based on the reduced control design model. However, note that the real power matrix P^R still inherently involves the evaluation model. It follows that, for a stable structure-control system the model efficiency becomes

$$e = \frac{x_o^T \mathbf{P}_c^m x_o}{x_o^T \mathbf{P}^R x_o}$$

(16)

Hence, the efficiency of the system in general depends on the initial disturbance state and the structure and control system parameters carried onto the power matrices via the Lyapunov Equations (13, 14). As simple as definition (16) of efficiency of the system appears, it does hold a host of internal information about the working of the structure and control system thereby characterizing the control/structure interactions uniquely as we outline below.

Since the control power matrices are Hennition matrices, the efficiency quotient (16) essentially represents a Rayleigh's quotient. Consider the eigenvalue problem associated with the power matrices (Ref. 2)

$$\mathbf{P}_c^m t_i = \lambda_i^e \mathbf{P}^R t_i \qquad i = 1, 2, \dots, 2n$$

(17)

where λ_i^e and t_i are defined as the ith characteristic efficiency and the ith controller efficiency mode, respectively. The eigenvector t_i is also referred to as the principal controller direction. Introducing the efficiency modal matrix T:

$$\mathbf{T} = [\, t_1 \quad t_2 \quad \dots \quad t_{2n} \,]$$

(18)

the following orthonormality relations can be stated

$$\mathbf{T}^T \mathbf{P}^R \mathbf{T} = \mathbf{I}_{2n \times 2n}, \qquad \mathbf{T}^T \mathbf{P}_c^m \mathbf{T} = \Lambda^e$$

(19)

where

$$\Lambda^e = \mathrm{diag}\,[\,\lambda_1^e \quad \lambda_2^e \quad \dots \quad \lambda_{2n}^e\,], \qquad \lambda_1^e \leq \dots \leq \lambda_{2n}^e \leq 1$$

(20)

From the properties of a Rayleigh's quotient, for any arbitrary vector (initial disturbance state) X_o. the value of the quotient (16) is bracketed by

$$\lambda_1^e \leq e \leq \lambda_{2n}^e \leq 1$$

(21)

where the upper bound 1 follows from the property (7). We shall refer to λ_1^ϵ as the fundamental efficiency. It is the minimum efficiency achievable by the structure-control system regardless of the initial state X_o.

3. Optimization Problem Formulation

In the design of structural-control systems it is natural to strive for a high modal efficiency e regardless of initial state disturbance. The consequence is that a high efficiency of any given reduced-order control design model will imply that there is low control power spillover to the truncated dynamics and hence minimized residual interaction with the design model. We can then pose a structure-control optimization problem which incorporates the system efficiency. The objective of the optimization problem is to minimize the total structural mass m subject to the constraints:

$$\omega_i \geq \omega_i^* \tag{23}$$

$$e\% \geq e^*\% \tag{24}$$

where ω_i is the closed-loop frequency and $e\%$ is the efficiency. An * denotes minimum desirable constraint values. From the definition of efficiency (16) it appears that the solution of the problem will also be sensitive to the initial modal state disturbance which is affected by the structural design variables. To circumvent this dependence of the problem solution on x_o we invoke a feature noted in Section 2 that the minimum efficiency achievable is the fundamental efficiency λ_1^ϵ regardless of the initial disturbance. Hence, the efficiency constraint (23) can be substituted by a constraint on the fundamental efficiency

$$\lambda_1^\epsilon > e^* \tag{25}$$

where sensitivity of λ_1^ϵ depends only on the system matrices via the efficiency eigenvalue problem (17). Hence, we solve the optimization problem subject to the constraints (22) and (23).

Sensitivities

The sensitivity expressions for the objective function and the frequencies ω_i are exactly the same as given in Ref. (4) where it is assumed that the control gain matrix G is the steady-state solution of the $2n^{th}$ order matrix Riccati equation associated with the minimization of the Control Design Performance Index (CDPI)

$$\text{CDPI} = \int_0^\infty \left(\underline{x}^T \mathbf{Q} \underline{x} + \underline{f}^T \mathbf{R} \underline{f} \right) dt \tag{26}$$

The only new sensitivity that is required here is the sensitivity expression of the fundamental efficiency λ_1^ϵ. From the controller efficiency eigenvalue problem (17), noting that P_C^M and P^R are symmetric positive definite, efficiency λ_i^ϵ sensitivities are

$$\lambda_{i,l} = t_i^T \left(\mathbf{P}_{c,l}^m - \lambda_i^\epsilon \mathbf{P}_{,l}^R \right) t_i \qquad i = 1, 2, \ldots, 2n \tag{27}$$

where l denotes partial derivative with respect to the lth design variable.

The sensitivities of the control power matrices are obtained from the Lyapunov equations (13, 14). Similar to the results of Ref. (4, 5), one obtains the following Lyapunov sensitivity equations:

$$\mathbf{A}_{cl}^T \mathbf{P}_{,l} + \mathbf{P}_{,l} \mathbf{A}_{cl} = -\hat{\mathbf{Q}}_{,l} \qquad \mathbf{P} = \mathbf{P}^R \text{ or } \mathbf{P}_c^m \tag{28}$$

where

$$\hat{\mathbf{Q}}_{,l} = \mathbf{Q}_{,l} + \mathbf{P} \mathbf{A}_{cl,l} + \mathbf{A}_{cl,l}^T \mathbf{P} \tag{29}$$

$$Q_{,l} = \left(GD^T M^{-1} D\right)_{,l} \text{ for } P = P^R \tag{30}$$

$$Q_{,l} = \left(GB^T BG\right)_{,l} \text{ for } P = P_c^m \tag{31}$$

for which the required sensitivities $A_{cl,l}, G, l, \Phi, l$ etc... are again given in Ref. (4). The sensitivity of M^{-1} is obtained from

$$M_{,l}^{-1} = -M^{-1} M_{,l} M^{-1} \tag{32}$$

4. Illustrative Example:

The ACOSS-FOUR structure shown in Fig. 1 was used to design a minimum weight structure with constraints on the closed-loop eigenvalues and the efficiency parameters. This structure has twelve degrees of freedom and four masses of two units each attached at nodes 1 through 4. The dimensions and the elastic properties of the structure are specified in consistent nondimensional units (Ref 4). Six colocated actuators and sensors are in six bipods. The controls approach used is linear quadratic regulator with constraint gain feedback. The weighting matrices for the state and control variables were assumed to be equal to identity matrices.

The nominal design is denoted by Design A with cross-sectional areas of the members equal to those given in Table 1. This nominal design weighs 43.69 units . This structure was initially analyzed with the first eight structural modes controlled.

The first two closed-loop frequencies ω_1 and ω_2 were found to be equal to 1.296 and 1.597, respectively. The efficiency parameter λ_1^c associated with the nominal design was found to be 0.407. The constraints imposed on the optimum design were as follows:

$$\omega_1 \geq 0.9069$$

$$\omega_2 \geq 1.117$$

$$\lambda_1^c \geq 0.549$$

The constraints specified on the closed-loop frequencies are 0.7 times those of the nominal design and the efficiency 1.35 times that of Design A. A constraint only on the efficiency would drive the lower frequencies to nearly zero during the optimization. Hence, they were constrained to 70% of the initial design. The initial design for optimization was Design A. The NEWSUMT-A software, based on extended interior penalty function method with Newton's method of unconstrained minimization was used to obtain an optimum design. After eight iterations the original structural weight of 43.69 units was reduced to 15.83 units. The optimum design was designated as Design B. The cross-sectional areas of the members, square of the structural frequencies and the closed-loop damping associated with two designs are given in Table 1-3.

5. Concluding Remarks

The concept of the efficiency of the structure-control system is used to design an optimum structure. The structural weight is assumed to be the objective function and the constraints are imposed on the closed-loop frequencies and the efficiency parameter. The illustrative example indicated that the structure–control efficiency of the reduced order system can be improved by using optimization techniques.

References:

1. Öz, H., Farag, K., and Venkayya, V. B. "Efficiency of Structure-Control Systems," *Journal of Guidance, Control and Dynamics.* Vol. 12, No. 3, May-June 1990.

2. Öz, H., "Efficiency Modes Analysis of Structure-Control Systems," AIAA - Paper No. 90-1210, Proceedings of the AIAA Dynamics Specialist Conference, April 1990, Long Beach, CA.

3. Khot. N. S., Öz, H., et al, "Optimal Structural Design with Control Gain Norm Constraint," AIAA Journal Vol 26, No. 5, May 1988, pp. 604-611.

4. Khot, N. S., "Structure/Control Optimization to Improve the Dynamic Response of Space Structures," *Computational Mechanics*, Vol. 3, 1988, pp. 179-186.

5. Schaechter, D. B., "Closed-Loop Performance Sensitivity to Parameter Variation," Journal of Guidance, Control, Vol. 6, 1983, pp. 399-402.
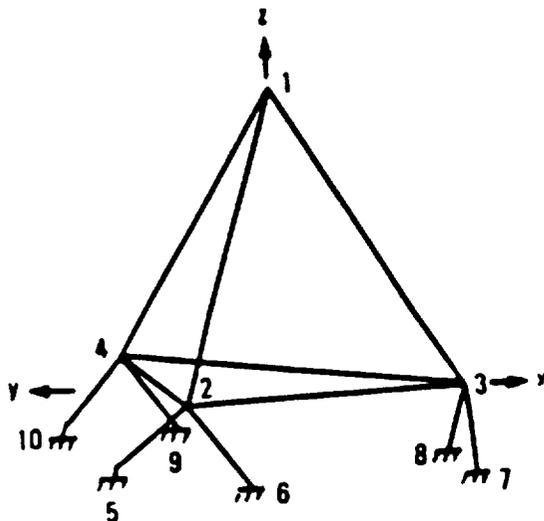
FIGURE 1: ACOSS-FOUR structure

Table 1.   Cross-Sectional Areas of the Members

Element No.	Design A	Design B
1	1000.0	678.62
2	1000.0	211.77
3	100.0	48.83
4	100.0	59.64
5	1000.0	70.13
6	1000.0	357.89
7	100.0	79.99
8	100.0	118.58
9	100.0	142.32
10	100.0	32.87
11	100.0	136.85
12	100.0	43.18
Weight	43.69	15.83
e%	40.7	54.9

Table 2.   Structural Frequencies (ω)

Mode	Design A	Design B
1	1.68	0.819
2	2.55	1.25
3	7.31	2.94
4	7.52	4.87
5	9.98	8.39
6	16.06	9.72
7	20.01	12.66
8	20.17	19.61
9	66.24	24.21
10	77.46	29.78
11	97.42	43.51
12	151.30	75.12

Table 3.   Closed Loop Damping

Mode	Design A	Design B
1	0.0563	0.0634
2	0.0674	0.1147
3	0.0739	0.9297
4	0.0805	0.0791
5	0.0848	0.1052
6	0.0866	0.0939
7	0.0762	0.0899
8	0.0718	0.0715

A Multidisciplinary Approach to Optimization
of Controlled Space Structures

Stanley E. Woodard and Sharon L. Padula
NASA Langley Research Center
Hampton, Virginia

Philip C. Graves
Vigyan Research Associates
Hampton, Virginia

Benjamin B. James
Lockheed Engineering and Sciences Company
Hampton, Virginia

A fundamental problem facing controls-structures analysts is a means of determining the trade-offs between structural design parameters and control design parameters in meeting some particular performance criteria. Developing a general optimization-based design methodology integrating the disciplines of structural dynamics and controls is a logical approach. The objective of this study is to develop such a method. Classical design methodology involves three phases. The first is structural optimization, wherein structural member sizes are varied to minimize structural mass, subject to open-loop frequency constraints. The next phase integrates control and structure design with control gains as additional design variables. The final phase is analysis of the "optimal" integrated design phase considering "real" actuators and "standard" member sizes. The control gains could be further optimized for fixed structure, and actuator saturation constraints could be imposed. However, such an approach does not take full advantage of opportunities to tailor the structure and control system design as one system.

The integrated optimization scheme used in this study is depicted in figure 1. An inner loop contains control analysis and structural analysis. The structural analysis produces mode slopes, frequencies, and the system inertia matrix. These are used by the control analysis to produce a value for the actuator mass based on required torque. This actuator mass value is then used in the structural analysis to regenerate the mode slopes, frequencies, and inertia matrix. These iterations continue until the mass of the actuator converges on a value relative to its prior value. Once a converged value for actuator mass is generated, structural and control gradients are estimated as uncoupled partial derivatives. The Generalized Sensitivity Equations use the uncoupled partial derivatives to approximate the appropriate coupled partial derivatives. These coupled partial derivatives are used by an optimizer to develop an optimal solution based on design objectives and constraints. Move limits are imposed so that the linear approximations to the controls-structures analysis remain valid. The outer loop is repeated until the contraints and objectives are met or until a predesignated number of cycles are completed.

A derivative of the Earth Observing System (EOS) is used for optimization studies. The reference model

is illustrated in figure 2. This model is an adequate representation of a "real world" application that can benefit from CSI-motivated trade-off. There are three structural design variables and twelve control design variables. The radius of the truss members, the radius of the antenna support trusses, and the radius of the antenna ribs are the structural design variables. Six position gains and six rate gains are used as the control design variables. The design objective is to minimize the total spacecraft mass (structure and actuator) while constraining vibration decay rate, that is:

$$\min[m_{actuator} + m_{structure}] \quad \text{while} \quad Re\{\lambda's\} \leq -\delta$$

where λ's are the system closed-loop eigenvalues. The performance question to be answered is whether structural mass can be traded for actuator mass.

Although with modern launch vehicles the total mass of a spacecraft similar to the one under consideration would not constitute a critical design driver, the actuator mass does play a key role in the entire system behavior. The actuator mass is the only design variable which is explicitly both a structural variable and a control variable. Because the actuator mass is non-negligible, it couples the open-loop finite element model with the control analysis, thereby requiring an iterative solution. The iterative process of computing actuator mass requires system mass matrix updating and recalculation of the eigensolution until the calculated actuator mass converges. The iteration is an "inner loop" within the optimization "outer loop."

The spacecraft's rigid-body motion and elastic behavior are used in the control and structural analysis. The analysis of the spacecraft's dynamics considers having the vehicle rotated from its initial attitude to a new attitude. After the rotation it is desired that any vibration that has occurred because of the maneuver be suppressed. The knowledge of the dynamics is used to size the controllers which are used to rotate the spacecraft and suppress its vibration. .

Torque wheels with bang-bang control slew the spacecraft during attitude maneuvers and, consequently, excite the structure. The model is slewed through some finite rotation. The maneuver is considered to be linear (i.e., small angular displacement over a long duration). In addition to the torque wheels used for the attitude maneuver, two collocated elastic controllers are located in the bays below the antenna supports. The modal representation of the elastic response, q, of the spacecraft due to the bang-bang maneuver and the collocated elastic controllers is governed by the following equation of motion:

$$\ddot{q} + D\dot{q} + \Lambda q = -\Psi^T G_p \Psi q - \Psi^T G_r \Psi \dot{q} + \Gamma^T M$$

Modal damping and stiffness are D and Λ, respectively. The first two terms on the right side of the equation above are the collocated elastic controller torques. These torques are proportional to the position gain, G_p, and the rate gain, G_r, and to the difference of mode slopes, Ψ, at the two elastic controller locations. The last term on the right-hand side of the equation above is due to the bang-bang control maneuver. This torque is proportional to the mode slope, Γ, at the point where the bang-bang torque is being applied and to the magnitude of the acceleration, M, produced by the torque. A state space representation of the system can be given as:

$$x = (q^T, \dot{q}^T)$$

$$\dot{x} = Ax + BM$$

$$A = \begin{pmatrix} 0 & I \\ [-\Lambda - \Psi^T G_r \Psi] & [-D - \Psi^T G_p \Psi] \end{pmatrix}$$

$$B = [0 \quad \Gamma^T]^T$$

where A and B are the system plant matrix and control input matrix, respectively. The elastic response can be calculated as:

$$x(t) = A^{-1}[e^{At} - I]BM \qquad 0 \le t \le t_f/2$$
$$x(t) = A^{-1}[e^{At} - 2e^{A(t-t_f/2)} + I]BM \qquad t_f/2 \le t \le t_f$$
$$x(t) = A^{-1}[e^{At} - 2e^{A(t-t_f/2)} + e^{A(t-t_f)}]BM \qquad t \ge t_f$$

Each of the two actuators produces the same torque output which is prescribed to be the product of gains, differential mode slope, and time response during the history of the spacecraft's motion. The torque output, u, for each actuator is resolved to three orthogonal directions according to the following relation:

$$u = -[G_p\Psi \quad G_r\Psi]x(t)$$

Total mass, m_a, for both actuators is proportional to the maximum torque magnitude along each direction ($u_{1max}, u_{2max}, u_{3max}$) such that:

$$m_a = 2m_t(u_{1max} + u_{2max} + u_{3max})$$

where m_t is a scaling factor for mass per unit torque.

Figure 3 illustrates the history of the elastic response norm, $\|x\|$. The attitude maneuver is a 10 second slew through 20 degrees. The peak elastic response occurs shortly after the switching time at 5 seconds. However, unless one would prefer to "rigidize" the spacecraft structure during the bang-bang maneuver, using a time response history after the end of the maneuver is sufficient. It is important to note that the peak response comes not at the end of the maneuver but shortly thereafter. Similarly, during the maneuver the peak response does not come at the switching time but shortly afterwards.

Because the actuator mass calculation depends upon the number of modes selected it is necessary to know which modes significantly contribute to the transfer function. If some of the major contributors are truncated or ignored, then the derived value of the actuator mass will be lower than necessary to control the flexible modes. As depicted in figure 4 the highest contribution to the transfer function occurs with the first 25 modes. Figures 5 and 6 show the torque needed in the x direction when 20 and 40 modes are used in the model, respectively. By omitting mode 23, the necessary control torque in the x direction was only half of what it should be.

Figures 7 and 8 show aggregate results of the overall optimization analysis. The design objective of minimizing total mass is illustrated in figure 7. Total mass converges at the twelfth cycle. The constraint on the design is that the real part of all closed loop eigenvalues be less than $\delta = -.014$. As depicted in figure 8, this constraint is satisfied after cycle 3 of the outer loop.

A method has been presented that combines structural analysis and control analysis integrated as a single optimization problem. This technique takes advantage of trade offs between control variables and structural variables to meet a performance objective. In the method illustrated the actuator mass value was prescribed to depend on the modal representation of the spacecraft motion. Modal selection has been shown to be a very important factor in determining the actuator mass. Knowing the elastic response was also a key factor. Aggregate results of the entire optimization loop have illustrated the effectiveness of using an integrated approach to meet performance objectives. Integrating the methods of control analysis and structural analysis for optimal performance will be a viable tool for the future design of flexible spacecraft.

503

Figure 2   Reference configuration.

Figure 1 Optimization with linearized and coupled analysis.

Figure 4 Modal contribution to transfer function.

Figure 3 Elastic response history.
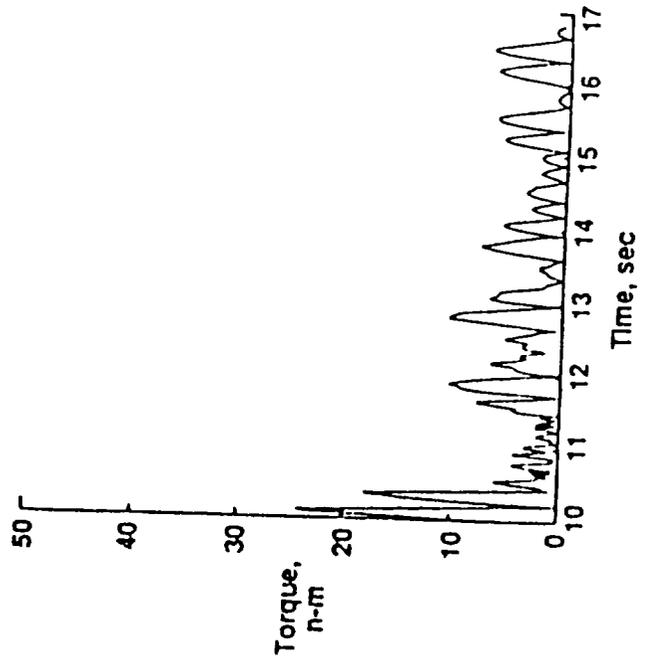
Figure 5 Control torque history. III - 20 modes
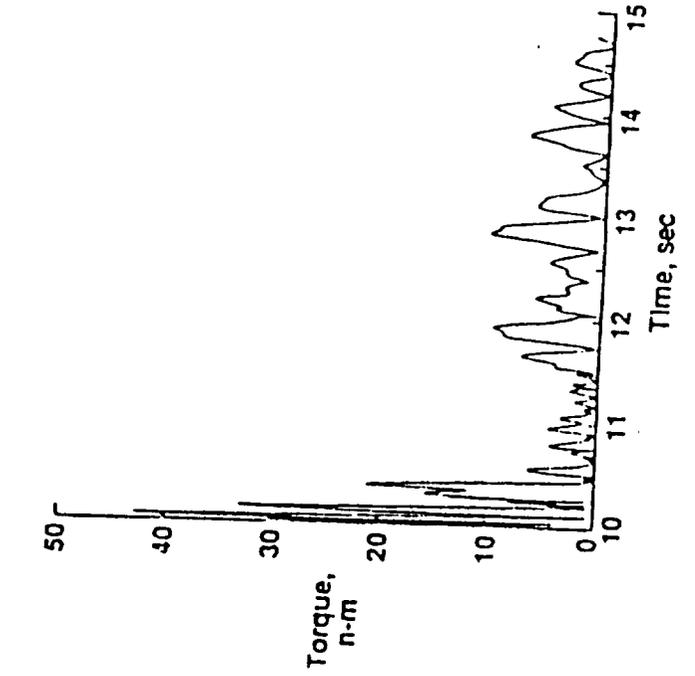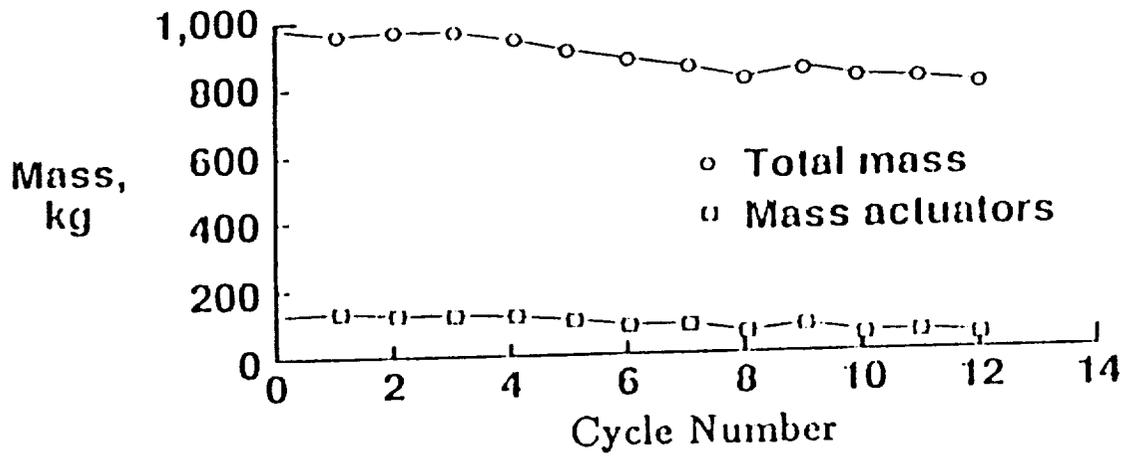
Figure 7 Mass convergence of slightly infeasible case using 15 design variables.
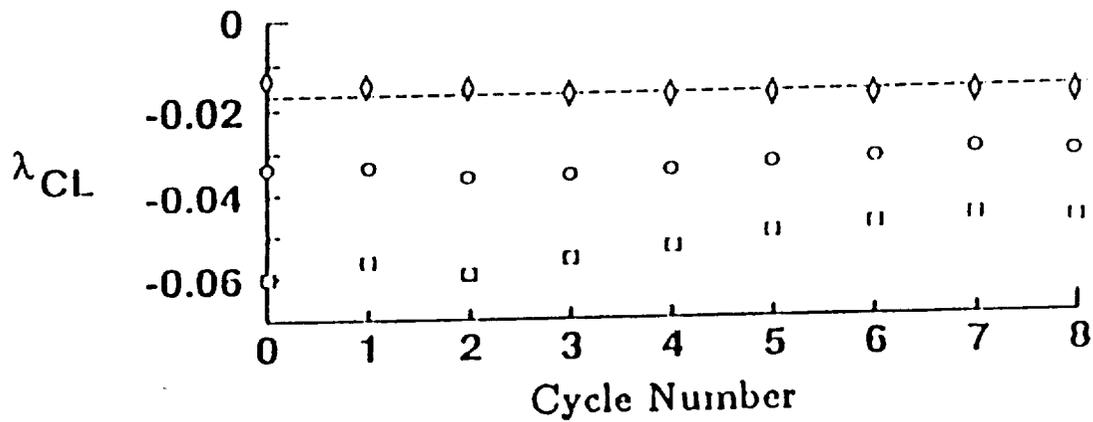
Eigenvalues

$\diamond\ \lambda_1 \quad \circ\ \lambda_3 \quad \Box\ \lambda_9$

Figure 8 Real part of closed-loop eigenvalues.

AN EFFICIENT DESIGN SENSITIVITY ANALYSIS OF EIGENVECTORS

T. TING

Department of Mechanical Engineering, University of Bridgeport, Bridgeport, Connecticut, U.S.A.

INTRODUCTION

Subspace iteration has been a major advance in solving large eigen problems when only a subset of eigen-pairs is required. The essence of this method is a transformation from displacement coordinates of an n-th order eigensystem to generalized coordinates of a smaller q-th order. The eigenvalue problem is then solved in the reduced space. The method was first developed by Clint and Jennings [1] for real symmetric systems and was then called "simultaneous iteration." The success of the method prompted further research along this line and there have been many improved algorithms developed [2,3]. This approach has been widely used by structural engineers for extracting the most useful natural frequencies and mode shapes of large-scale dynamic systems.

Recently, the increasing importance of automated modification for dynamic system has brought on the interest in developing efficient method for computing the design sensitivities of large scale dynamic system's eigen-parameters. The method for determination of eigenvalue derivatives is almost conclusive and is very simple and straightforward. The real problem arises in calculating eigenvector derivatives which is found to be much more complicated. Fox and Kapoor's pioneering work [4] in 1968 has laid down two primary directions for developing computational methods to calculate eigenvector derivatives of discrete systems. Their first approach was derived based on the first-order variation of a single eigenvalue equilibrium equation and its eigenvector mass normalization equation. Thus, this requires only the specific eigenvalue and eigenvector and can be termed the direct approach. Nelson [5] presented a method in 1976 which simplified calculation along this line and the method is well received as one of the best methods available for the purpose. However, since Nelson's method still requires to solve a $n \times n$ set of equations for each eigenvector, it becomes a costly process when the derivatives of a large number of eigenvectors, provided n is large, are demanded. Fox and Kapoor's second approach employed modal space expansion concept where the eigenvector derivative spans the entire modal space for the exact solution. They also proposed an approximated solution by spanning a subset of the modal space. The former requires the complete set of eigen-pairs which is prohibitively expensive for large systems. Although the later only requires a relatively smaller set of eigen-pairs, the inaccuracy of the approximation may present a problem and there is still no clear guideline on how a subset of eigen-pairs can be selected to approximate the exact with an acceptable tolerance. However, if the derivatives of a large number of eigenvector is required, this approach may become a preferred method owing to its simplicity in computation.

This paper exploits into a new direction which is in the form of iterative process for simultaneously calculating eigenvector derivatives of many eigenvectors with respect to multi-variables. The method fully use all the available information from preceding eigenvalue solution and, thus, effectively economizes computational efforts. It iterates through two equations derived from the first variation of the two fundamental equations used in subspace iteration method. There is no expensive large matrix decomposition required and the process converges to acceptable solution in a finite number of iterations. Therefore, the procedure increases its efficiency superiority over the others as the system size or the number of interested eigenvectors become larger and larger.

SUBSPACE ITERATION FOR EIGEN-PROBLEMS

Suppose we are interested in extracting the lowest p, $p<n$, eigenvalues and corresponding eigenvectors of a n-th order system, the solution to the reduced problem can be written in matrix form

$$\mathbf{K\Phi} = \mathbf{M\Phi\Lambda} \tag{1}$$

where $\mathbf{\Phi}$ is the modal matrix of $n \times p$ containing the required eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix of order p with the eigenvalues on its diagonal. Let us now consider the shifted subspace iteration described by

$$(\mathbf{K} + \sigma\mathbf{M})\overline{\mathbf{U}}^{(k+1)} = \mathbf{M}\mathbf{U}^{(k)} \tag{2}$$

where σ is a well chosen shift value and $k = 1,2,...$, is the iteration counter. $\mathbf{U}^{(k)}$ is a $n \times q$ ($p<q<n$) matrix and, provided, whose columns are M-orthonormal. Solving Eq.(2) yields $\mathbf{U}^{(k+1)}$. The next step is using $\mathbf{U}^{(k+1)}$ as projection vectors to project $\mathbf{K}$ and $\mathbf{M}$ into matrices of $q \times q$ and forms the reduced eigenvalue problem

$$\mathbf{K}^{(k+1)}\mathbf{Q}^{(k+1)} = \mathbf{M}^{(k+1)}\mathbf{Q}^{(k+1)}\mathbf{\Lambda}^{(k+1)} \tag{3}$$

where

$$\mathbf{K}^{(k+1)} = \left(\overline{\mathbf{U}}^{(k+1)}\right)^T \mathbf{K}\overline{\mathbf{U}}^{(k+1)} \tag{4}$$

$$\mathbf{M}^{(k+1)} = \left(\overline{\mathbf{U}}^{(k+1)}\right)^T \mathbf{M}\overline{\mathbf{U}}^{(k+1)} \tag{5}$$

are $q \times q$ real symmetric matrices. The solution of the reduced eigen-problem of Eq.(3) yields $\mathbf{\Lambda}^{(k+1)}$ and $\mathbf{Q}^{(k+1)}$. Then, for the next iteration, we shall use

$$\mathbf{U}^{(k+1)} = \overline{\mathbf{U}}^{(k+1)}\mathbf{Q}^{(k+1)} \tag{6}$$

To summarize and for simplification reasons, let

$$\mathbf{P} = \mathbf{\Lambda} + \sigma\mathbf{I} \tag{7}$$

and

$$\mathbf{W} = \mathbf{\Phi}\mathbf{P}^{-1} \tag{8}$$

The two fundamental equations for the shifted subspace iteration at convergence can be written as

$$(\mathbf{K} + \sigma\mathbf{M})\mathbf{W} = \mathbf{M}\mathbf{\Phi} \tag{9}$$

$$\widetilde{\mathbf{K}}\mathbf{P} = \widetilde{\mathbf{M}}\mathbf{P}\mathbf{\Lambda} \tag{10}$$

where

$$\widetilde{\mathbf{K}} = \mathbf{W}^T\mathbf{K}\mathbf{W} \tag{11}$$

and

$$\widetilde{\mathbf{M}} = \mathbf{W}^T\mathbf{M}\mathbf{W} \tag{12}$$

Assuming eigenvectors Φ are available and form an M-orthonormal basis of p-dimensional subspace of the operators K and M, we want to determine the first derivatives of Φ with respect to design variables $x = \{x_1, x_2, ..., x_m\}$. It is also provided that all the eigenvectors in Φ are associated with simple eigenvalues so that Φ is continuous and differentiable with respect to x. Since all the eigenvalues are simple and distinct, their first derivatives with respect to x can be uniquely determined.

Now, taking the partial derivative of both sides of Eqs.(9) and (10) with respect to design variable x_j and rearranging yields

$$(K + \sigma M)\frac{\partial W}{\partial x_j} = \frac{\partial M}{\partial x_j}\Phi - (\frac{\partial K}{\partial x_j} + \sigma \frac{\partial M}{\partial x_j})W + M\frac{\partial \Phi}{\partial x_j} \tag{13}$$

$$\widetilde{K}\frac{\partial P}{\partial x_j} - \widetilde{M}\frac{\partial P}{\partial x_j}\Lambda = \frac{\partial \widetilde{M}}{\partial x_j}P\Lambda + \widetilde{M}P\frac{\partial \Lambda}{\partial x_j} - \frac{\partial \widetilde{K}}{\partial x_j}P \tag{14}$$

where

$$\frac{\partial \widetilde{K}}{\partial x_j} = \frac{\partial W^T}{\partial x_j}KW + W^T\frac{\partial K}{\partial x_j}W + W^TK\frac{\partial W}{\partial x_j} \tag{15}$$

$$\frac{\partial \widetilde{M}}{\partial x_j} = \frac{\partial W^T}{\partial x_j}MW + W^T\frac{\partial M}{\partial x_j}W + W^TM\frac{\partial W}{\partial x_j} \tag{16}$$

The relation between $\dfrac{\partial W}{\partial x_j}$ and $\dfrac{\partial \Phi}{\partial x_j}$ can also be derived form Eq.(8) that

$$\frac{\partial \Phi}{\partial x_j} = \frac{\partial W}{\partial x_j}P + W\frac{\partial P}{\partial x_j} \tag{17}$$

Altogether, Eqs.(13) - (17) can be used to construct recurrence relations for an iterative process of computing eigenvector derivatives.

The following algorithm is an subspace iteration scheme for calculating eigenvector derivatives. The algorithm begins with a set of trial vectors for $\partial\Phi/\partial x_j$ and updates the trial vectors based on the recurrence relations to improved the approximation.

Let $\Psi^{(1)}$ be the set of initial trial vectors. For $k=1,2,...,$ the recurrence relations are

$$(K + \sigma M)V^{(k+1)} = \overline{F} + M\Psi^{(k)} \tag{18}$$

where

$$\overline{F} = \frac{\partial M}{\partial x_j}\Phi - (\frac{\partial K}{\partial x_j} + \sigma \frac{\partial M}{\partial x_j})W \tag{19}$$

is constant throughout the iterations. Solve for $V^{(k+1)}$ in Eq.(18) and employ Eqs.(15) and (16) to compute

$$\frac{\partial \widetilde{K}^{(k+1)}}{\partial x_j} = W^T \frac{\partial K}{\partial x_j} W + W^T K V^{(k+1)} + (W^T K V^{(k+1)})^T \qquad (20)$$

$$\frac{\partial \widetilde{M}^{(k+1)}}{\partial x_j} = W^T \frac{\partial M}{\partial x_j} W + W^T M V^{(k+1)} + (W^T M V^{(k+1)})^T \qquad (21)$$

Next, consider the following relation

$$\widetilde{K} D^{(k+1)} - \widetilde{M} D^{(k+1)} \Lambda = \widetilde{M} P \frac{\partial \Lambda}{\partial x_j} + \frac{\partial \widetilde{M}^{(k+1)}}{\partial x_j} P \Lambda - \frac{\partial \widetilde{K}^{(k+1)}}{\partial x_j} P \qquad (22)$$

and solve for $D^{(k+1)}$ using a Nelson-like method. Then, for the next iteration, we shall use the updated trial vectors

$$\Psi^{(k+1)} = V^{(k+1)} P + W D^{(k+1)} \qquad (23)$$

The iteration process converges, so that

$$\Psi^{(k+1)} \rightarrow \frac{\partial \Phi}{\partial x_j} \qquad \text{as} \qquad k \rightarrow \infty \qquad (24)$$

CONVERGENCE RATE AND ITERATION VECTORS

The ultimate rate of convergence of iteration vectors to the derivatives of the i-th eigenvector, $\partial \phi_i / \partial x_j$, is $\lambda_i / \lambda_{q+1}$, where q is the number of the lowest eigenvectors used in the iteration process. A large q, $q > p$, is normally required in order to achieve a good convergence rate. But this also increases the computational effort within each iteration for carrying the iteration vectors associated with the extra eigenvectors. However, it can be shown that if the lowest p eigenvectors were computed accurately enough, Eq.(23) for updating the iteration vectors becomes uncoupled for those associated with the lowest p eigenvectors at the second iteration. This implies that

$$D^{(2)} = \begin{bmatrix} D_{pp} & D_{pq} \\ D_{qp} & D_{qq} \end{bmatrix}^{(2)} \qquad (25)$$

and $[D_{pp}]$ is a diagonal matrix of order p×m. This means that extra iteration vectors are only necessary at the first iteration to speed up the convergence and they need not be included in the subsequent iterations.

It is also interesting to note that, usually, the method does not require good starting trial vectors for improving the overall rate of convergence. In fact, most of generated initial trial vectors, including the truncated modal approximation, will result in the same trial vectors after the first iteration. Therefore, for convenience, it may be adequate to assign the initial trial vectors the null vectors, i.e., $\Psi^{(1)} = 0$.

AN OVERRELAXED SCHEME

It is a demonstrated fact that overrelaxation can significantly improve the convergence characteristics of the subspace iteration method for eigenvalue analysis. The overrelaxation for the subspace iterations of eigenvalue keeps all equations the same as its basic version except to update the new iteration vectors $U^{(k+1)}$ by

$$U^{(k+1)} = U^{(k)} + \left(\overline{U}^{(k+1)}Q^{(k+1)} - U^{(k)}\right)\alpha$$

(26)

where α is a diagonal matrix with its diagonal terms equal to individual vector overrelaxation factors $\alpha_i, i = 1,...,q$. Referring to Bathe's analysis [6], we have

$$\alpha_i = \frac{1}{1 - (\lambda_i + \sigma)/(\lambda_{q+1} + \sigma)}$$

(27)

Similarly, the subspace iterations for eigenvector derivatives can be overrelaxed by letting

$$\Psi^{(k+1)} = \overline{\Psi}^{(k)} + \left((V^{(k+1)}P + WD^{(k+1)}) - \Psi^{(k)}\right)\overline{\alpha}$$

(28)

where $\overline{\alpha}$ is the overrelaxation factor matrix with diagonal elements $\overline{\alpha}_j, j = 1,...,p\times m$. Since the left hand side matrix of Eq.(18) is the same as that of its original equation for eigenvalue problems, Eq.(9), the eigen-properties associated with the solution of this equation remain the same. Therefore, if the j-th iteration vector is a trial vector for one of the derivative vectors of eigenvector ϕ_i, we have

$$\overline{\alpha}_j = \alpha_i$$

(29)

The overrelaxation should be employed after the iteration vectors have settled down. Normally, applying overrelaxation once every three iterations should be effective.

The overrelaxation on the solution of Eq.(18) can reduce the number of iterations required to obtain an acceptable result. On the other hand, the elimination of the need to solve for $D^{(k+1)}$ in Eq.(22) can drastically reduce the computing cost within each iteration. Theoretically, the iteration process converges when k+1 approaches ∞ and

$$[D]^{(\infty)} = \left[\frac{\partial P}{\partial x_j}\right] = \left[\frac{\partial \Lambda}{\partial x_j}\right]$$

(30)

However, this can be achieved in two iterations with the provision that the given lowest p eigenvectors are exact. In the case of using inaccurate eigenvectors in the subspace iterations, the matrix $D^{(k+1)}$ will never be diagonal even after many iterations. However, for adequately accurate eigenvectors, the matrix $D^{(k+1)}$ become near-diagonal after two iterations where the diagonal teams almost equal to their theoretical values and the off-diagonal terms are, relatively, much smaller. Hence, it seems to be reasonable to approximate $D^{(k+1)}$ by Eq.(30) after the first iteration. It should also be award that when the quality of the eigenvectors is poor this approximation may lead to a divergence. The remedy to the problem is to compute the true $D^{(k+1)}$ once every several, say three, iterations.

EXAMPLE PROBLEM

The FE model of a power turbine blade subject to fixed end condition is used to demonstrate the effectiveness of the subspace iterations for eigenvector derivatives. The model consists of 200 flat plate elements comprising 231 grid points as shown in Fig.1. The elements are divided into 20 rows, from the root to the tip, of 10 elements each.

A single design variable is defined as a scale factor for all the element thicknesses in the fifth row from the root. The efficiency of the subspace iterations for computing the eigenvector derivatives with respect to the design variable has been compared against that of Nelson's method. Both Nelson's and subspace iteration methods have been implemented in MSC/NASTRAN's DMAP [7] and some modification to the previous DMAP version of Nelson's method [8] has been done to improve its machine-dependent efficiency.

Tables 1 and 2 summarize the comparison over CPU time for calculating derivatives of six and twelve eigenvectors, respectively. This example was run on DEC VAX/785-11 and the CPU seconds shown in the Tables are the standard VAX CPU seconds.

The convergence criterion for terminating the subspace iterations is based on the error norms defined by

$$\varepsilon_i = \frac{\left| \phi_i^{(k+1)} - \phi_i^{(k)} \right|}{\left| \phi_i^{(k+1)} \right|} \tag{31}$$

where $|\cdot|$ denotes the vector norm of any kind. In this example, the Euclidean vector norm is used and the tolerance for ε_i is set to be $5.\times10^{-4}$ (i.e., $\varepsilon_i \le 5.\times10^{-4}$ $\forall i = 1,...p\times m$).
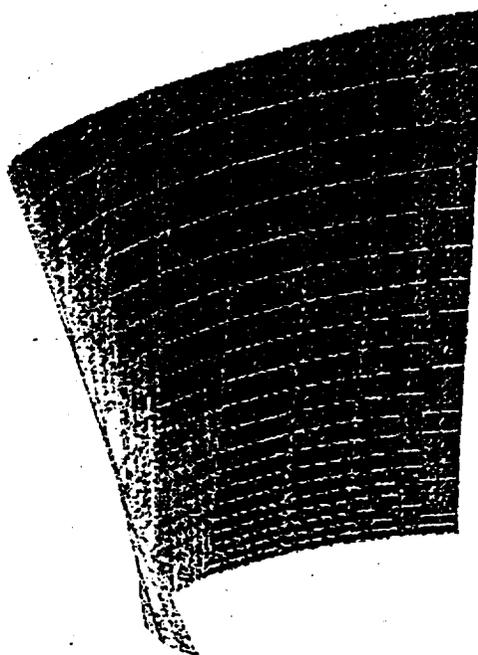
Fig.1. Power turbine blade FE model

Table 1 Efficiency comparison for 6 modes

	BASIC SUBSPACE ITERATIONS			RELAXED	NELSON'S
p*/q**	6/6	6/12	12/12	6/12	—
No. of Iterations	12	5	5	4	—
CPU seconds	539.7	307.3	494.	244.5	667.5

* Number of iteration vectors after the first iteration.
** Number of iteration vectors at the first iteration.

Table 2. Efficiency comparison for 12 modes

	BASIC SUBSPACE ITERATIONS			RELAXED	NELSON'S
p*/q**	12/12	12/20	20/20	12/20	—
No. of Iterations	16	7	7	6	—
CPU seconds	1181.7	622.1	1004.	492.9	1293.6

* Number of iteration vectors after the first iteration.
** Number of iteration vectors at the first iteration.

CONCLUSIONS

This paper has presented the basic and an overrelaxed subspace iteration methods for calculating eigenvector derivatives of general real eigen-systems. The solution algorithms have been implemented, and the results of a sample problem are reported. The basic formulation is directly derived from the equations of the basic subspace iterations for solving eigenvalue problems. The overrelaxation of the subspace iterations shares the result of Bathe's analysis for eigenvalue problem. This is due to the fact that the eigen-properties of the solution equations are the same for both purposes. In fact, since the placements of eigenvalues are known *a priori*, Bathe's overrelaxation formula can be easily employed without having the difficulty of estimating eigenvalues for the evaluation of the overrelaxation factors.

The subspace iterations for eigenvector derivatives does not require the decomposition of a system-size matrix. Thus, this approach is desirable for very large systems where decomposition of the system matrix may cause spilling operation to occur which results in prohibitively high costs. Even for a moderately large system, as shown in example, the basic algorithm can achieve 20-50% reduction in CPU and the overrelaxation algorithm gains more than 60% of saving in CPU. Nevertheless, since Nelson's method does not require much computational effort for additional design variables, the subspace iterations may loss its superiority over Nelson's method for sufficiently large number of design variables.

In the normal circumstances, the convergence rate of this approach does not seem to be affected by the selection of initial trial vectors. The quality of the given eigenvectors may influence the convergence rate. However, it is worthy of mentioning that the coupled basic subspace iterations converge to the true eigenvector derivatives regardless of the quality of the eigenvectors provided. On the other hand, the accuracy of Nelson's results is directly determined by the quality of the eigenvectors used in calculation.

REFERENCES

1. M. Clint and A. Jennings, 'The evaluation of eigenvalues and eigenvectors of real symmetric matrices by simultaneous iteration', *Computer J.*, 13, 76-80 (1970).

2. K-J. Bathe and E.L. Wilson, 'Large eigenvalue problems in dynamic analysis', *J. Engng Mech.Div., Proc. ASCE*, EM6, 1471-1485 (1972).

3. K.-J. Bathe and S. Ramaswamy, 'An accelerated subspace iteration method', *Comp. Meth in App. Mech and Engng.*, 23, 313-331 (1980).

4. R.L. Fox and M.P. Kapoor, 'Rates of change of eigenvalues and eigenvectors', *AIAA J.*, 6(12), 2426-2429 (1968).

5. R.B. Nelson, 'Simplified calculation of eigenvector derivatives', *AIAA J.*, 1(9), 1201-1205 (1976).

6. K.J. Bathe, 'Convergence of subspace iteration', in: *Formulations and Numerical Algorithms in Finite Element Analysis*, MIT Press, Cambridge, MA, (1977).

7. M.A. Gockel, et al (eds.), *MSC/NASTRAN User's Manual*, II, The MacNeal Schwendler Corporation, Los Angeles, CA, (1985).

8. T. Ting and D. Wallerstein, 'Calculation of eigenvector derivatives in design sensitivity analysis', *MSC/NASTRAN Application Manual*, Application Note, The MacNeal Schwendler Corporation, Los Angeles, CA, (1986).

Expert System for Multidisciplinary Analysis and Optimization Using ASTROS

Hari K. Sarma [1], Ramana V. Grandhi[2]

Department of Mechanical and Materials Engineering

Wright State University

Dayton, Ohio 45435

and

Ronald F. Taylor[3]

University of Dayton Research Institute

Dayton, Ohio 45469

---

[1] Graduate Research Assistant
[2] Associate Professor
[3] Research Engineer

Abstract

This paper discusses the framework of a Knowledge Based Expert System (KBES) environment to design aerospace structures under structural and aerodynamic constraints using ASTROS (Automated Structural Optimization program). ASTROS is a synthesis tool built around the NASTRAN finite element program. The knowledge base capabilities are discussed for synthesizing in statics, normal mode, steady and unsteady aerodynamic disciplines. A description of the two ASTROS advisor modules the Editor/Bulk Data generator and Post-processor is included. Experiences and issues involved in hierarchical representation of knowledge as menu options at different levels of abstraction are presented. A brief overview of Knowledge Based Systems and the shell CLIPS (C Language Integrated Production System) used to develop the ASTROS Advisor are discussed. Illustrative examples of the advisor in designing airframe structures are also included.

Introduction

Today engineers make extensive use of computers in engineering synthesis. The iterative processes involved in design and analysis have been implemented as well-defined algorithmic procedures. As numerical procedures and associated computer codes become more and more complex, engineers are starting to use Knowledge Based Systems to simplify the task of implementing these sophisticated numerical tools. By applying the technology of Expert Systems, engineers are able to communicate parts of the ill-structured design knowledge through a computer medium. Some examples of ill-structured knowledge in multidisciplinary optimization deal with the establishment of the initial parameters, selection of important constraints, identification of design objectives, and selection of design methodologies.

In the past, several prototype knowledge based systems have been developed for various aerospace and civil engineering structures.

SACON [1] (Structural Analysis Consultant) is one of the earliest expert systems developed for structural analysis. It recommends the type of analysis strategy required for MARC which is a large general purpose finite element analysis program. Rivlin, et al [2] developed a similar consultant for MARC that creates the input file to run an elastic, elasto-plastic, creep, dynamic or large displacement analysis using beam, quadrilateral or brick elements. Buckling Expert [3] integrates an expert system shell with three external algorithmic programs (an optimization code, an analysis code and a relational data base manager) using generic interface routines to design stiffened cylindrical composite panels and shells. Hajela [4] developed an expert system framework to aid the users of the finite element analysis program EAL (Engineering Analysis Language) in nodes selection, mesh generation, and element selection. STRUTEX [5] which interfaces algorithmic procedure for structural analysis and design with expert systems developed on two different expert system shells aids a structural engineer to initially configure a structure to support point loads. In the field of numerical structural optimization, EXADS [6] aids a novice user to select an appropriate optimization strategy from ADS (Automated Design System). IDESIGN [7] used heuristical knowledge to identify active constraint sets, feasible and infeasible designs, choice of algorithm based on convergence criteria, classification of problems as linear, nonlinear etc.

The impact of Knowledge Based Systems on multidisciplinary aerospace structural design is studied herein by discussing the framework and implementation of a prototype Knowledge Based System developed for ASTROS, a finite element based synthesis program which includes both structural and aerodynamic constraints. The multidisciplinary knowledge involved in structural analysis, aerodynamic analysis and numerical optimization are discussed. Experiences and issues in the development of an intelligent environment utilizing a menu type editor module (Advisor) and a post-processor consultant module are presented. Integration of these independent modules is done within the framework of CLIPS (C Language Integrated Production System) [8], a computer language /Expert System shell developed by

the Artificial Intelligence Section (AIS) at NASA/Johnson Space Center. Knowledge is represented as facts and production rules in CLIPS which employs a forward chaining inference engine. CLIPS provides high portability, low cost and easy integration with external systems. An independent bulk-data sub-module within the advisor module generates analysis and design models for both steady and unsteady aerodynamics disciplines in the format required by ASTROS. The generated aerodynamic bulk data model is appended to the existing finite element models. The advisor has features to automatically check the syntax and hierarchical data structure. It also has provisions to modify the model at different abstraction levels. The Advisor is available in IBM PCs, Macintosh, VAX and Sun Workstation versions.

ASTROS ADVISOR

Knowledge acquisition, evaluation and validation are critical elements in the development of expert systems. These are intensive procedures involving both people and time. The development of an expert system for a multidisciplinary design and analysis code like ASTROS, expertise from diverse subsets of knowledge domains is required. Fig. 1 shows the unique approach adopted in this Expert system development. In this approach a third entity, a user specialist is added. The user specialists are multiple experts in the field of finite elements, optimization, aerodynamics, and an experienced ASTROS user with background knowledge of AI. The user specialists use knowledge presented in the ASTROS manuals, heuristical experiences gained from running several problems on ASTROS, and expertise in a particular discipline to evaluate the prototype system at each stage. The prototype expert system is evaluated for knowledge representation and user acceptance issues. Based on the feedback from user specialist and evaluations from the domain expert, the prototype expert system is continously refined by the knowledge engineer.

System Architecture

The ASTROS Advisory System contains two independent knowledge based systems, an ASTROS Advisor and a Post-processor Consultant, integrated within the same CLIPS architecture. A high level command language program directs the user to either of these modules. The Advisor comprises of two independent sub-modules, an intelligent Editor and a Bulk Data Generator (BDG). They can be selected independently or sequentially one after another. They can be used to either create or modify ASTROS input files. For existing ASCII ASTROS input files the keywords are scanned and the data is converted to CLIPS facts. The Advisor is structured as directed menus with relevant options and advice on how to select these options. The menu interface also allows access to any element of the ASTROS input file. This is handled in CLIPS by asserting a fact to trigger the rule that handles the display of the menu.

The ASTROS Advisor Editor has a main menu which provides the user with an option to either create or modify the Executive Control and the Solution Control sections. The Optimize and Analyze subsections of the Solution Control are made up of menus to define Boundaries, Disciplines and Discipline-Options in a descending order. To execute the Bulk Data Generator (BDG) module, an existing solution control file with either the Steady Aero or Flutter discipline is required. The BDG module is made up of four modules: (i) aerodynamic model panel and configuration generator, (ii) a boundary parameter generator, (iii) an unsteady parameter generator and (iv) a steady parameter generator. The user can select either the default values provided in the panel data menu or input his/her choice of values. All input values are checked for modelling accuracy by heuristics, and a corrective action is initiated by the system for any needed modifications. By default, the paneling data generated by the BDG module is shared by both the steady and unsteady aerodynamic model wing panels if they coexist. However, independent panel data can be generated by using the toggle options in the wing panel menu. The Unsteady and Steady Aero menus contain default values based on standard reference data used in aerodynamics or data based on heuristical experience. These include standard air density reference value, flight symmetry conditions

517

for steady aerodynamic discipline and an ideal number of reduced frequencies for the flutter discipline. User input in terms of simple flight conditions like altitude, Mach number, etc.. are requested by the BDG for any missing fields to generate the appropriate performance bulk data cards. Options are available to use conversion factors to maintain consistency of input units. The aerodynamic bulk data file created is appended to the solution control file. As shown in Fig. 2, the BDG creates the aerodynamic model and links it to an existing finite element model at user defined grid points. The finite element structural model is later appended as an external file by the user.

The Post-processor has been developed as a traditional Production rule system. A set of allowable default boolean values make up the intial fact base. The system is modular with additional knowledge entered in the form of questions, identified problems and corresponding solutions. A simple question and answer session with the consultant establishes a possible error in the design problem, and correspondingly the advisor provides a solution. The remember and recall features of the Post-processor consultant allows a user to interrupt an interaction with the system, terminates the session, and saves as a restart file. The knowledge is abstracted hierarchically down with additional facts solicited only after evaluating user response. The system provides explanation features for its reasoning thereby making the knowledge transparent. Help capabilities are provided by informing the user where to look for additional assistance in answering a particular query. The Post-processor provides recommendations which are useful in formulating and successfully running the future design models.

Illustrative Examples

The multidisciplinary swept wing model [9] is selected to demonstrate the modelling capabilities of the ASTROS Advisor. Creation of the wing model for steady aerodynamics and flutter disciplines under two different boundary conditions are demonstrated in this illustration. The first boundary condition idealizes the wing supported for plunge modes at center root of the structural box only and the second boundary condition idealizes the wing cantilevered at the root. Fig. 3 shows a typical menu display for Executive and Solution Control Editor. The editor also checks and updates the input model completeness and syntax of the executive and solution control decks before the user can exit from the module. Fig. 4 shows a typical menu display in the Bulk Data Generator modules. The wing panel geometry is shared by both the Steady and Unsteady Aerodynamics disciplines. However, the distribution pattern of the aerodynamic boxes is different for the two disciplines which is achieved by changing the toggle setting in the Specific Wing Panel to single for just that menu value (distribution for chordwise boxes). The menu options required to generate data for the wing configuration (airfoil data) are displayed only for the steady aero model. User input of mach number, altitude, flutter velocity limit, reduced frequencies and default values for symmetry present in the Unsteady Aero Menu are used to generate data specifying the flutter conditions for the wing. Density ratios are generated by the system based on input altitude and default reference density. The range of velocities to be included for the analysis are generated by the system based on flutter velocity limit and heuristical empirical formulas. User input of mach number, velocity, altitude, load factor and default value for symmetry present in the Steady Aero Menu are used to generate data specifying the symmetric trim conditions for the wing. The dynamic pressure is calculated by the system based on air density, altitude and flight velocity. The trim type 1 is inferred by the system based on its knowledge of symmetric condition (if symxz=1, then trmty =1).The pitch rate is calculated from user input of load factor and velocity. Fig. 5 shows the input file created by the advisor based on the information provided in the editor and bulk data generator modules. Only a portion of the bulk data cards generated for this example problem are shown for clarity. This input file generated by the Advisor has comment cards in the bulk data.

Fig. 6 lists an interactive session with the Postprocessor consultant which provides advice for a non-converged optimization run. The help features present in the Postprocessor are

also illustrated. Help is requested for understanding the queries about approximate problem convergence and constraint violations. A separate ADVICE file is also generated containing the Post-processor Advice.

Conclusions

The framework of a knowledge based expert system environment (ASTROS Advisory System) to design aerospace structures under structural and aerodynamic constraints was demonstrated. Experiences and issues in the development of an intelligent environment consisting of a menu type editor and a post-processor consultant were presented. An independent bulk-data sub-module within the editor module generated analysis models for both steady and unsteady aerodynamics disciplines in the format required by ASTROS. CLIPS proved to be a viable tool for expert system development with minor modification. The ASTROS Advisory System is available on IBM compatible PC's, Macintosh and VAX computers.

Acknowledgements

References

1 Bannet, J.S. and Engelmore, R.S. "SACON: A Knowledge Based Consultant for Structural Analysis" *Proceedings of Sixth International Joint Conference on Artificial Intelligence, Tokyo, Japan*, vol 1, 1979, pp 47-49.

2 Rivlin, J.M. Hsu, M.B. and Marcal, P.V. " Knowledge Based Consultant for Finite Element Structural Analysis " AFWAL TR-80-3069, Wright Patterson Air Force Base, Ohio, (May 1980).

3 Zumsteg, J.R. and Flaggs, D.L. "Knowledge Based Analysis and Design Systems for Aerospace Structures" *Proceedings of the 1985 ASME Winter Annual Meeting, Miami Florida,* vol 1, Nov 1985, pp 67-78.

4 Hajela, P. " Expert Systems in Structural Modelling and Design " *NATO Advanced Study Institute, Computer Aided Optimal Design, Portugal,* vol 3, July 1986, pp 384-393.

5 Rogers, J.L. Sobieszczanski, J. and Feyock, S. "STRUTEX: A Prototype Knowledge-based System For Initially Configuring a Structure To Point Loads In Two Dimensions " NASA Technical Memorandum 100613, (April 1988).

6 Rogers, J.L. Jr and Barthelemy, J.M. "An Expert System for Choosing the Best Combination of Options in a General-Purpose Program for Automated Design Synthesis " *Engineering with Computers,* vol 1, No: 4, 1986, pp 217-227.

7 Arora, J.S. and Baenziger, G. "Uses of Artificial Intelligence in Design Optimization " *Computer Methods in Applied Mechanics and Engineering,* vol 54, 1986, pp 303-323.

8 Giarratano J. and Riley G., Expert Systems Principles and Programming, PWS-Kent Publishers, Boston, 1989.

9 Johnson, E.H. and Neill, D.J. ASTROS Applications Manual, AFWAL-TR-88-3028, Wright Patterson Air Force Base, Ohio (1988).
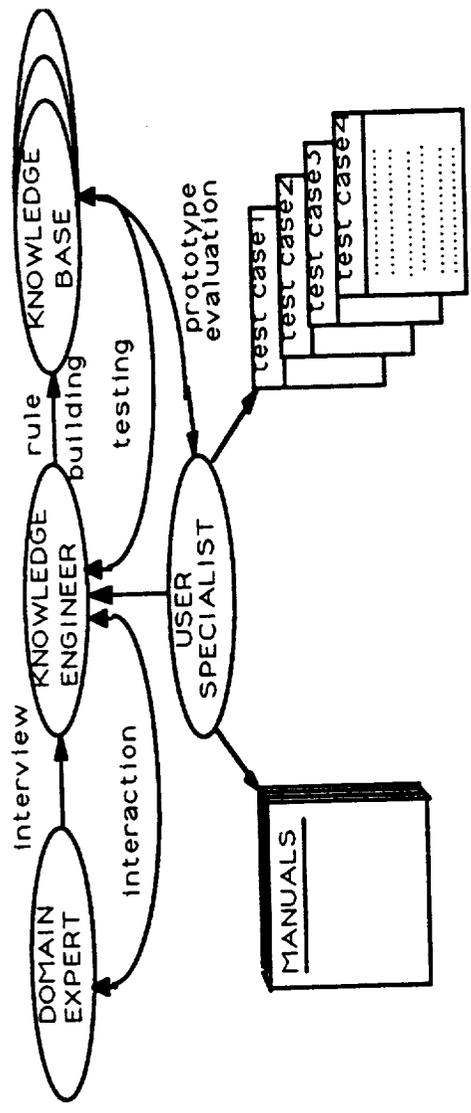
Fig. 1 ASTOS ADVISOR Development
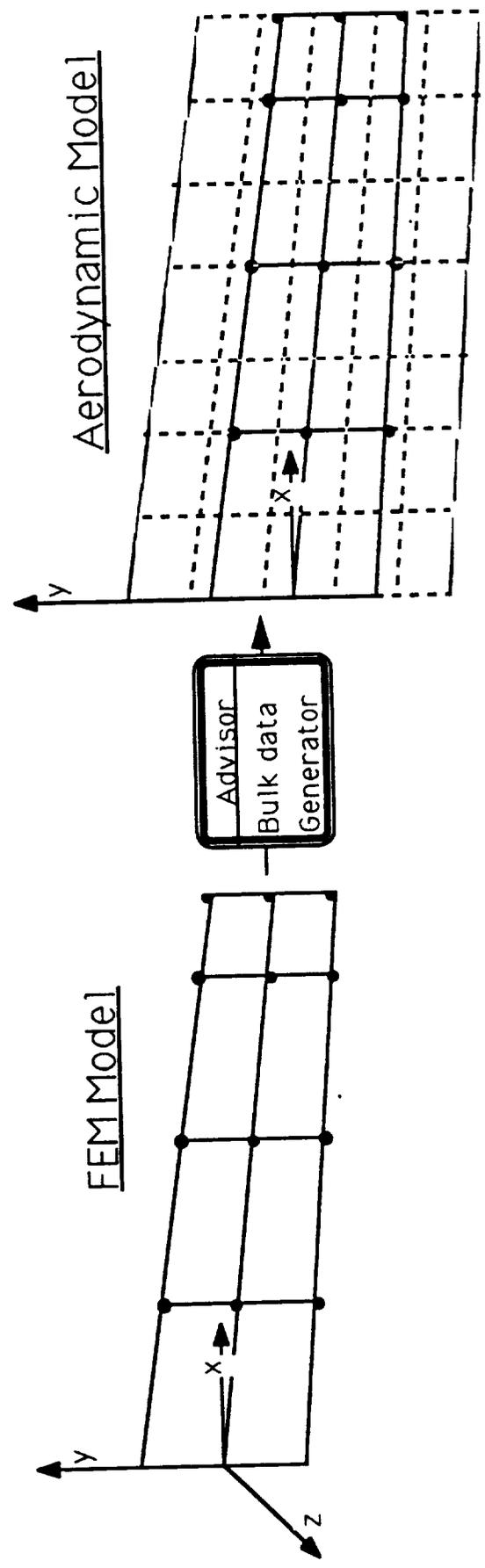
Fig 2 Bulk Data Generator Capabilities

<1> Execution & Solution Control Editor
<2> Bulk Data Generation Parameters Editor
<3> EXIT ASTROS ADVISOR

>1

1. CREATE a NEW Solution Control file
2. MODIFY OLD Solution Control file

>1

Enter the name of the run-time data base file
that is to be loaded.   (8 A/N)

>mydb

Enter the password for the database file.   (8 A/N)

>mypass

EXECUTIVE & SOLUTION CONTROL EDITOR
MAIN MENU

Make your menu/edit selection by typing the code
contained within the symbols: <>.

DataBase Filename   : MYDB

<1> EXECUTIVE CONTROL Section
<2> OPTIMIZE Section
<3> ANALYZE Section
<E> EXIT & SAVE CHANGES to control deck
<Q> QUIT & DON'T SAVE CHANGES to control deck

-1->3

Fig.3 Screen Display of ASTROS Advisor Main Menu

BULK DATA GENERATOR
AERODYNAMIC MODELS MENU

<1> WING Panel Definition
-<2> HORIZONTAL TAIL Panel Definition
-<3> VERTICAL TAIL Panel Definition
-<4> BODY Panel Definition
-<5> POD Panel Definition

<Return> RETURN to Bulk Data Generator MAIN MENU

>1

WING PANEL DEFINITION

MODEL. . . . . : steady_aero
TOGGLE SETTING . . . . . : BOTH
TIP-TO-TIP SPAN. . . . . : 0
WING AREA. . . . . : 0
ASPECT RATIO . . . . . . : 0
<F> ROOT OFFSET FROM CENTER. : 0

EXISTING PANELS
*** empty list ***

<A> ADD a trapezoidal panel
<D #> DELETE a trapezoidal panel

<V> VIEW other model
<C> CHANGE update mode to work on THIS MODEL ONLY
<RETURN> Return to PANEL MENU

>a

PANEL DATA

MODEL   : unsteady_aero
PANEL #   : 1
PANEL AREA: 1

<1> Spanwise Panel Length . . . . : 1
<2> Leading Edge Sweep Angle. . . : 0 (degrees)
<3> Dihedral Angle. . . . . : 0 (degrees)

<4> Length of Chord at the Root . : 1
<5> Length of Chord at the Tip. . : 1

<6> Chordwise Distribution. . . . : COSINE
<7> Number of Chordwise Boxes . . : 4

<8> Spanwise Distribution . . . . : SINE
<9> Number of Spanwise Boxes. . . : 4

<Return> Return to Bulk Data Generation Main Menu

Fig.4 Screen Display of ASTROS Advisor Wing Panel Menu

ASTROS INPUT DATA FILE
———————————

ASSIGN DATABASE MYDB MYPASS NEW DELETE
SOLUTION
TITLE = Multidisciplinary Swept Wing Model
ANALYZE
  SUBTITLE = Aerodynamic Analysis Model
  PRINT DISP = ALL
  BOUNDARY MPC = 100, SPC = 110, REDUCE = 120, METHOD = 999
    FLUTTER (FLCOND = 45)
    LABEL = Unsteady Flutter Analysis Subsonic
    PRINT ROOT = ALL
  BOUNDARY MPC = 101, SPC = 111, REDUCE = 121, SUPORT = 1000
    SAERO (TRIM = 19)
    LABEL = Static Aero Analysis Supersonic
    PRINT TRIM

END
BEGIN BULK
INCLUDE aeromodel

Fig. 5 Input File Generated by ASTROS Advisor

        AERO MODEL BULK DATA FILE
        —————————————

$**
$ AEROS ACSID RCSID REFC REFB REFS GREF REFD REFL
$
AEROS, , ,172.15, 1400.000, 241010., ?, 0.0
$**
$ AERO ACSID REFC RHOREF
$
AERO, ,172.15, 1.1463e-07
$ CONVERT VELOCITY V
CONVERT VELOCITY V

CONVERT, VELOCITY, 19.9444
$**
$ FLUTTER SID   METHOD DENS MACH VEL MLIST EPS -- CONT
$ CONT   SYMXZ SYMXY
$
FLUTTER, 45, PK, 145, 0.8, 345, , , , +bc0
+bc0, 1, 0
$———— density ————
$ FLFACT SID F1 F2 F3 F4 F5 F6 F7 CONT
$ CONT   F8 F9 F10 -ETC-
$
FLFACT, 145, 1
$———— velocity ————
$ FLFACT SID F1 F2 F3 F4 F5 F6 F7 CONT
$ CONT   F8 F9 F10 -ETC-
$
FLFACT, 345, 265, 371, 450.5, 477, 530, 583
$————————— unsteady aero panel #1—————————
$ CAERO1 EID PID CP NSPAN NCHORD LSPAN LCHORD IGID CONT
$ CONT   X1 Y1 Z1 X12 X4 Y4 Z4 X43

$********************* ASTROS POST PROCESSING ADVISOR *********************
$
$ This is an advisory tool to be used when an optimization study
$ has been performed and there are concerns regarding the solution.
$ It will try to identify the problem and give suggestions for a solution.
$
$ VALID RESPONSES: yes no y n YES NO Y N why help h halt exit
$
$ (EXIT: Allows you to leave the Post Processor with the ability to restart
$ from the point where you left off.
$ HALT: Allows you to stop the execution of CLIPS rules in order to view
$ the FACTS, with the capability of restarting the rule execution
$ by typing in 'RUN'.)
$
$**

Do you want to:
  1. Start the session from the beginning, or
  2. Continue where the last session left off?
>1

"Has ASTROS converged?"

>n

"Has the approximate problem converged?"

>n

"Are constraints being violated in any iteration?"

>h

HELP ---> "Violated constraints are those constraint values
    which are positive in the Summary of Active Constraints
    Table. This table is printed when the DCONS option is
    in your print command."

"Are constraints being violated in any iteration?"

>n

"Were approximate constraints violated in the last iteration?"

>y

PROBLEM -> "The move limit may be too lenient."

These are the prioritized solutions:

SOLUTION -> "Decrease MOVLIM < 2 (but > 1)."

SOLUTION -> "If shape functions were used, try physical linking
(use DESELM or DESVARP bulk data entries instead of DESVARS)."

Fig. 6 Sample listing of Postprocessor session

Development of a Knowledge-Based System for Validating Finite Element Models

N.I. Munir and J.N. Kudva

Structural Methods and Applications

Northrop Corporation, Hawthorne, California

ABSTRACT

The finite element modeling of an airframe structure requires knowledge of general purpose programs such as NASTRAN as well as a detailed understanding of the airframe structure. Due to the sophistication of general purpose programs such as NASTRAN, a substantial investment in time and effort is required to gain expertise in using them effectively. This paper describes the development of an expert system used in the validation of NASTRAN based finite element models.

Experts in the NASTRAN based finite element modeling of airframe structures were interviewed to document, understand and represent their knowledge and reasoning in the expert system. Finite element stress analysis and internal loads reports generated by the experts were reviewed to determine expert resolution of problem areas. As a result, areas requiring expert assistance in the modeling of the airframe structures were identified.

The finite element input data is represented as a set of 'facts'.

A rule based representation is used to code the expert knowledge. A hierarchical set of rules are applied. The expert system first acts as an intelligent front end to insure that all the pre-requisites needed to perform the analysis are present. This includes material properties, boundary conditions and connectivity information. The next step examines if incompatible sets of elements are connected. The succeeding step examines if the specific airframe component is modeled by the appropriate set of elements. The next and final step examines if the airframe members used are adequate to represent the anticipated state of stress. The expert system is designed to inform the user the severity of the error, the likely consequence and possible remedial action.

The shell used in the development of the expert system is CLIPS. CLIPS contains a forward chaining inference engine based on the Rete algorithm. CLIPS may be implemented on most personal computers as well as mini computers and mainframes. The approach taken, however is general and can be implemented using commercially available expert system shells.

Introduction

NASTRAN is a general purpose analysis code that brings together the state of the art analysis capabilities into a single program for the analysis of complex structures. The usefulness of NASTRAN and other similar codes in predicting the structural response depends on a large part on a proper idealization and discretization of the structure. The process of structural idealization and discretization is referred to as the generation of a finite element model. The finite element model, for a given structure is not unique. Several finite element models, each yielding acceptable solution accuracy may be constructed for the same structure. This paper will focus on the use of a Knowledge Based System (KBS) in aiding the user to validate finite element models constructed for airframe analysis. Finite element model validation is achieved by insuring that airframe-specific modeling guidelines, coded into rules have been satisfactorily met.

Figure 1 shows the taxonomy for combining the KBS with NASTRAN. The KBS sits between the analysis package and the user. The KBS uses as input a user generated NASTRAN bulk data file. The KBS interprets the input file, interactively asks the user additional information, applies airframe specific modeling rules and generates an output file. This file records any violation in the modeling of airframe structures and reports it to the user before commitment to a potentially expensive analysis.

In the capacity mentioned above, the KBS serves as an Intelligent Front End (IFE). The IFE serves to validate the analysis by checking the input finite element analysis input data using airframe specific modeling guidelines and general finite element analysis modeling guidelines. Anomalies, if any are reported to the user prior to commitment to a potentially expensive analysis. The major steps in the development of the KBS along with an application example are described next.

```
┌─────────────────┐
│  NASTRAN        │
│  INPUT DATA     │
└────────┬────────┘
         │
┌────────┴────────┐
│  KNOWLEDGE      │
│  BASED          │
│  SYSTEM         │
└────────┬────────┘
         │
┌────────┴────────┐
│     USER        │
└─────────────────┘
```
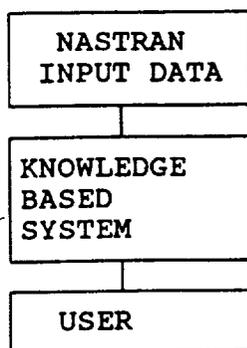
Figure 1: Interaction Between User, NASTRAN and Knowledge Based System

Knowledge Acquisition

The knowledge acquisition consisted of obtaining procedures and methods used in the NASTRAN modeling of aircraft structures. Three principal sources of knowledge were : (i) Structural analysis guidelines (ii) Stress analysis reports and (iii) Interviews with engineers experienced in the modeling of airframe structures using NASTRAN.

The structural analysis guidelines provide analysis procedures for some of the more complicated problems encountered in vehicle structural analysis. Information obtained from the guidelines allowed for the selection of an appropriate class of problems that are amenable to solution using knowledge based methods.

The stress analysis reports provide detailed information about the airframe finite element models. General modeling principles common to airframe components were identified. Also, the modeling approaches used to meet specific analysis objectives were examined. Generalizations obtained from examination of the stress analysis reports were later included in the knowledge base.

Engineers experienced in the NASTRAN modeling of airframe structures were interviewed. As a result, areas where engineering judgement was consistently used to either develop or interpret the airframe model were identified. Also identified were areas where inexperienced NASTRAN users may most likely make errors in the modeling of airframe structures. An important feature that could be readily accommodated using knowledge based techniques was the need for the ready identification of the airframe finite element model in terms of airframe members (For instance: Element No. 60042 BAR STRINGER). In meetings with experienced engineers it was decided that the KBS will be best suited for use with internal load models.

Development Tool

For the internal load models considered, a rule based network can be used to effectively represent the expert knowledge. New data can then be introduced as a set of 'facts'. These facts typically act on the IF portion of the rules and result in some specific actions. The actions may be the generation of additional facts. These facts can then fire other rules. This process continues until all rules whose LHS match with available facts have been fired. Other necessary requirements are (i) forward chaining inference mechanism and (ii) the ability to interface with FORTRAN analysis routines.

The C Language Integrated Production System, CLIPS was selected as the tool to develop this expert system. CLIPS is a rule-based forward chaining tool. CLIPS has the required software hooks to interface with both the operating system and other application programs. The software hooks are in the form of C Language interfaces. The availability of the CLIPS source code allows CLIPS to be compiled and implemented in most installations that support the C Language.

Description of Knowledge-Based-System

The components of the KBS consist of a Data-Base Management System (DBMS), analysis routines and the rules used by the inference engine. The DBMS facility is needed to reduce the user supplied analysis data to manageable records of similar entities. The FORTRAN analysis routines use the user supplied input data to calculate additional entities that aid in the classification of the airframe components. The rules are applied to the input data and additional related data generated by the analysis routines to complete the classification task and validate the finite element model. The three components are described next.

NASTRAN allows an arbitrary input sequence. For efficient data manipulation similar entities need to be grouped and identified. This grouping allows the KBS to access needed data on demand. To achieve this, a DBMS capable of reading the NASTRAN bulk data and storing the input data by NASTRAN card identification symbols is required. For instance, entities such as grid identification numbers and coordinates, element type and connectivity information need to be stored in a form where selective and easy retrieval is possible. CADDB, the database management facility linked to ASTROS was used. Any, similar facility may be used to organize the NASTRAN input data.

The analysis routines compute parameters that aid in the classification of structural components. These routines use the NASTRAN input data stored in text files by the DBMS. The routines, also query the user interactively for additional information. The user is prompted to enter information relating to the orientation of airframe skins, spars and webs. The FORTRAN routines are used to compute the direction cosines of the surface elements. The direction cosines of these elements are checked against the orientations of the airframe structural components input by the user. This is used to classify the surface elements. The classification for each surface element used is skin, spar or rib. Elements that do not satisfy the criteria for any of the airframe members are stored in a special file for later retrieval by the user.

Rules are then applied that identify other airframe components. STRINGERS are identified next. These are structural members modeled by either BAR or ROD elements, having a cross-sectional area greater than 0.001 square inches and connected to a SKIN airframe member identified earlier. A SPAR_CAP is a STRINGER having nodes in common with a SPAR element. Similarly, a RIB_CAP is a STRINGER having nodes in common with a RIB. This completes the identification of airframe members for wing-type structures. The list of elements not identified are stored in a file for later retrieval.

The second task of the KBS is to examine the airframe model and determine if suitable elements have been selected to model the airframe. Rules for the modeling of airframe components are invoked. Two factors are generally considered in deciding the suitability of modeling a specific airframe member (i) if the element selected can represent the stresses adequately and (ii) if displacement compatibility is maintained at the element common nodal points. Elements not meeting either of the two criteria will be reported to the user. Appropriate element types to model the airframe component will be suggested to the user. This final task is of assistance mainly to engineers inexperienced in the modeling of airframe structures, and may be omitted by the more experienced engineers.

An example of the application of the KBS to an aircraft wing is shown on page 6. First, the classification task is performed and second, the modeling of the airframe members is examined to determine if the appropriate finite elements have been used. Airframe members modeled incorrectly are reported to the user.

Cost Savings

For experienced engineers the classification task of the KBS will be of assistance in quickly identifying airframe members. This classification task, particularly the identification of non-structural members can result in substantial time savings in interpreting models that are not used frequently. Since the KBS is to be implemented in a production environment a direct comparison of the time savings due to the classification capability of the KBS could not be determined. Estimates of time savings in interpreting the finite element model in terms of airframe structural components range from 60 to 70 percent.

Engineers inexperienced in airframe analysis will benefit from the modeling critique provided by the KBS. This may be used as a training tool for inexperienced engineers. It has the potential of reducing the learning time by allowing the engineers to learn from production type models as opposed to text-book or artificially contrived models.

```
                    ┌─────────────────┐
                    │ NASTRAN         │
                    │ BULK DATA       │
                    └─────────────────┘
                             │
                    ┌─────────────────┐
                    │    D B M S      │
                    └─────────────────┘
                             │                    ┌──────────────────┐
                             │                    │ DIR'N COSINES:   │
                    ┌─────────────────┐           │ SKIN, SPARS      │
                    │ FORTRAN ANALYSIS│───────────│ RIBS             │
                    │    ROUTINES     │           ├──────────────────┤
                    └─────────────────┘           │  USER INPUT      │
                             │                    └──────────────────┘
                    ┌─────────────────┐
                    │    C L I P S    │
                    │ CLASSIFICATION RULES│
                    └─────────────────┘
```

CAPS | SKIN | SPAR | RIB

RIB CAPS

SPAR CAPS
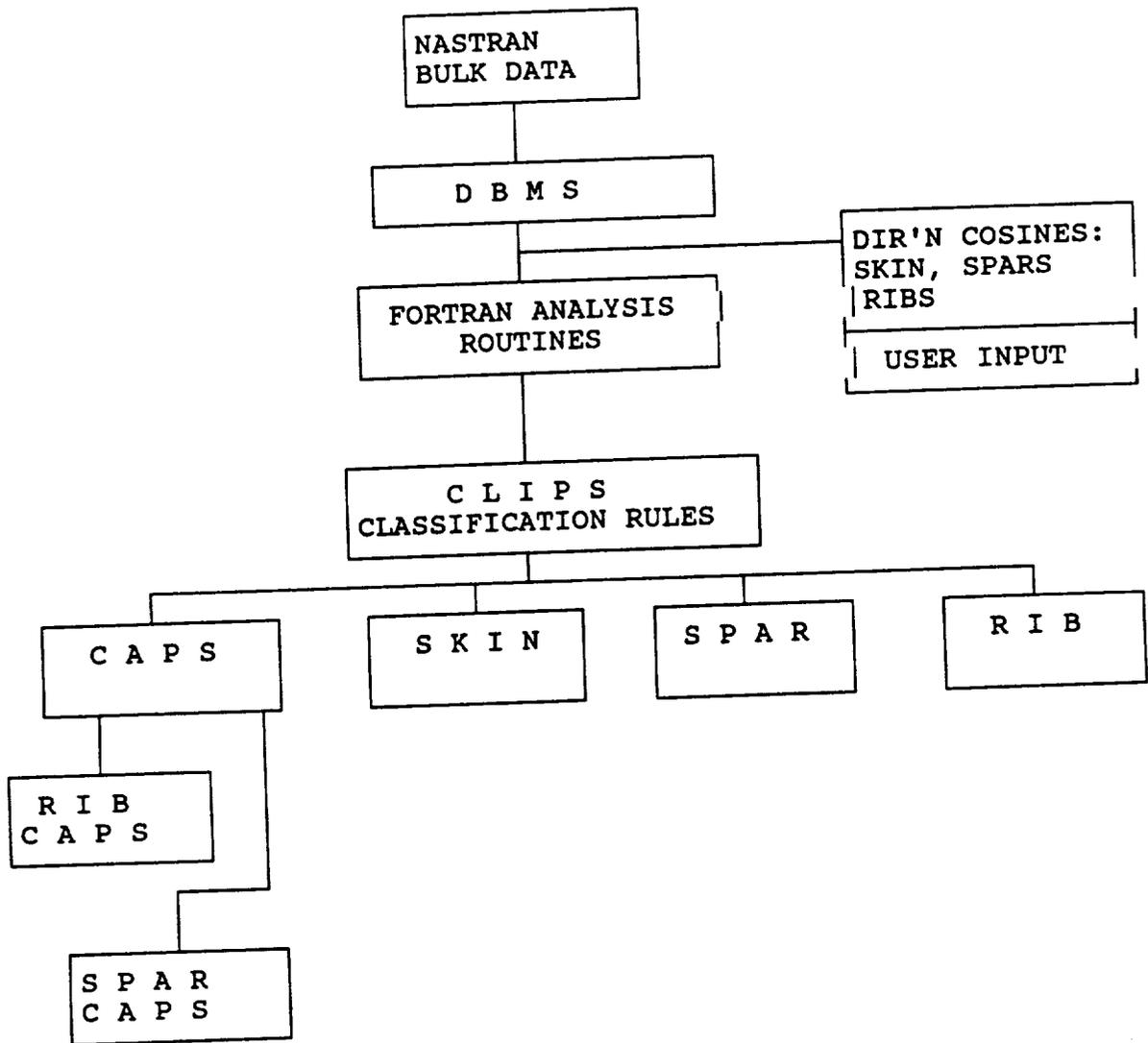
CLIPS: AIRFRAME MODELING RULES

OUTPUT

Typical Output:
*** WARNING ****

AF: STRINGER   MODELED BY: CBAR   54006   NODE1: 34281 NODE2: 34283
AF: SKIN   MODELED BY: CSHEAR 32642   IG1: 342612 IG2: 342618
IG3: 342624   IG4: 342628

Partial Output and Classification of an Aircraft Wing
Structure Using the KBS

References

1. CLIPS Reference Manual Version 4.3 Johnson Space Center June 1989, Report No. JSC-22948.

2. Rose, T., "Practical Finite Element Modeling Techniques Using MSC/NASTRAN", MacNeal-Schwendler Corporation, Seminar Notes, March 1989.

3. Cameron, S.J., "FEM Modeling Guidelines", Northrop Structural Design Manual (in preparation).

4. Howard, H.C., Wang, J., Daube, F., Rafiq, T., "Applying Design-Dependent Knowledge in Structural Engineering Design", AI-EDAM , Vol. 3, No. 2, pp. 111-123

5. Johnson, E.H., and Neill, D.J., "Automated Structural Optimization System (ASTROS)", Final Report, AFWAL-TR-88-3028, July 1988

On a Concurrent Element-by-Element Preconditioned Conjugate Gradient Algorithm for Multiple Load Cases

by

Brian Watson and M. P. Kamat
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332

N94-71491

Abstract

Element-by-element preconditioned conjugate gradient (EBE-PCG) algorithms have been advocated for use in parallel/vector processing environments as being superior to the conventional LDL^T decomposition algorithm for single load cases. Although there may be some advantages in using such algorithms for a single load case, when it comes to situations involving multiple load cases, the LDL^T decomposition algorithm would appear to be decidely more cost-effective. The authors have outlined an EBE-PCG algorithm suitable for multiple load cases and compared its effectiveness to the highly efficient LDL^T decomposition scheme. The proposed algorithm offers almost no advantages over the LDL^T algorithm for the linear problems investigated on the Alliant FX/8. However, there may be some merit in the algorithm in solving nonlinear problems with load incrementation, but that remains to be investigated.

Introduction

A conjugate gradient algorithm for solving a linear system of equations, or equivalently for minimizing a function of several variables, is often encountered in structural optimization and analysis problems. Using a finite element discretization scheme, problems of solid mechanics may be reduced to a set of linear equations or equivalently to minimizing the potential energy expressed as a function (quadratic for linear problems) of the nodal displacement degrees of freedom. The need to solve finite element models with large numbers of degrees of freedom has created the need for highly efficient algorithms for solution.

Modern single-processing computers are limited to an estimated maximum performance of 3 billion floating point operations per second (3 gigaflops). However, recent advances in computer architectures provide the means to achieve higher performance ratings through the use of parallel and/or vector processing. Multiple instruction, multiple data (MIMD) computers, which allow several processors to operate independently on different sets of data, are considered to have the greatest potential for increased performance ratings.

In a sequential computing environment, the system of linear equations are often solved using the very well known LDL^T decomposition of the associated stiffness matrix followed by simple forward and backward substitutions. However, this technique involves inherently sequential operations and thus its potential on a multiprocessor is limited.

The conjugate gradient algorithm is an iterative method which involves developing a set of mutually conjugate directions and performing successive 1-dimensional minimizations in each of these directions. It has been proven[1] that, using exact arithmetic, the conjugate gradient algorithm will converge to the minimizer of a quadratic in no more than n iterations for an n d.o.f system. This feature makes the conjugate gradient algorithm attractive among iterative procedures. Additionally, the conjugate gradient algorithm, especially as proposed by Law[2], appears to be nicely suited to a parallel/vector computing environment.

Conjugate Gradient Algorithm

The formulation of the total potential energy function, Π, for a structural system typically results in an expression of the form:

$$\Pi = \frac{1}{2}\{q\}^T[K]\{q\} - \{q\}^T\{F\} + \{b\} \qquad (1)$$

where:

$\{q\}$ is the vector of unknown nodal displacements

$[K]$ is the symmetric, positive definite structure stiffness matrix

$\{F\}$ is the vector of externally applied nodal forces

$\{b\}$ is an arbitrary constant vector

The conjugate gradient algorithm for determining the minimizer for expression (1) is given below in Figure 1.

Select $\{q\}_0$

$\{r\}_0 = \{F\} - [K]\{q\}_0$

$\gamma_0 = \{r\}_0^T\{r\}_0$

$\{p\}_0 = \{r\}_0$

Begin Loop: i=1

$\{u\}_i = [K]\{p\}_{i-1}$

$\alpha_i = \dfrac{\gamma_{i-1}}{\{p\}_{i-1}^T\{u\}_i}$

$\{q\}_i = \{q\}_{i-1} + \alpha_i\{p\}_{i-1}$

$\{r\}_i = \{r\}_{i-1} - \alpha_i\{u\}_i$

$\gamma_i = \{r\}_i^T\{r\}_i$

$\beta_i = \dfrac{\gamma_i}{\gamma_{i-1}}$

$\{p\}_i = \{r\}_i + \beta_i\{p\}_{i-1}$

i -> i+1 : Repeat Loop

Figure 1. The conjugate gradient algorithm for the minimization of a quadratic function.

531

The residual vectors, $\{r\}_i$, are the negative of the gradient of the total potential energy function at each $\{q\}_i$. The vectors, $\{p\}_i$, are the $[K]$-conjugate search directions. The algorithm terminates after the mth iteration when the magnitude of the residual, $\{r\}_m$, is small compared to the magnitude of the applied load vector, $\{F\}$ (i.e. $\frac{\gamma_m}{\{F\}^T\{F\}} <$ Tolerance).

The most expensive computation in the conjugate gradient algorithm is the calculation of the matrix-vector product, $[K]\{p\}_i = \{u\}_i$. This calculation is well suited to a parallel computing environment in that it can be accomplished at the element level. Thus, an element-by-element conjugate gradient algorithm allows each processor to perform the independent calculation, $[K]^e \{p\}_i^e = \{u\}_i^e$, where $[K]^e$ is the element stiffness matrix and $\{p\}_i^e$ is the element search direction. Then, the vector $\{u\}_i$ is computed by an assemblage of the $\{u\}_i^e$. The remaining computations are all well suited to vector processing.

A diagonal preconditioning matrix, $[T]$, can be applied to the original system to improve the condition number of the stiffness matrix[3]. By introducing $[k]=[T]^T[K][T]$, $\{y\}=[T]^{-1}\{q\}$, $\{f\}=[T]^T\{F\}$, and applying the conjugate gradient method to the function, $\pi = \frac{1}{2}\{y\}^T[k]\{y\} - \{y\}^T\{f\} + \{B\}$, the solution may be determined in fewer iterations. Once the minimizing vector, $\{y\}^*$, of the new function, π, is determined, the minimizing vector, $\{q\}^*$, of the original function, Π, can be calculated from the transformation, $\{q\} = [T]\{y\}$.

More complicated preconditioning matrices have been proposed. An example is the element-by-element preconditioning scheme by Hughes et al.[4]. However, for this problem, it is believed that the additional computational effort required in the implementation of these preconditioners is likely to outweigh any potential advantage resulting from the reduction of iterations. Thus, only the diagonal preconditioner has been investigated.

Single Load Case

The element-by-element conjugate gradient algorithm was implemented on an Alliant FX/8 parallel/vector supercomputer. Limited success in performance increases over the solution subroutine COLSOL[5] has been achieved. COLSOL uses the LDL^T decomposition of the stiffness matrix to solve the linear system of equations $[K]\{q\} = \{F\}$.

The test problem used was generated from a finite element approximation to a flat plate subjected to transverse concentrated loads using thin plate elements (See Figure 2.). By adjusting the mesh size, problems with different numbers of unknowns were generated. Additionally, adjusting the node numbering sequence allowed different bandwidths of the stiffness matrix, $[K]$ (See Table I.).

Simply Supported - All Sides
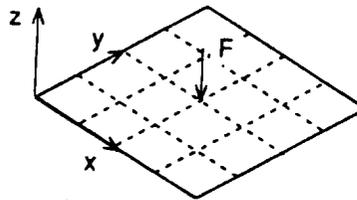
Figure 2. Test problem model

Table I. Statistics for Finite Element Models.

Mesh	Number of Unknowns	Half-Bandwidth Numbering Scheme 1	Half-Bandwidth Numbering Scheme 2
10 x 10	279	35	271
20 x 20	1159	65	1141
30 x 30	2639	95	2611

Figure 3. shows the time required for the element-by-element conjugate gradient method (EBE-PCG) and COLSOL for solving the various problems. Note that the performance of the EBE-PCG is independent of the bandwidth of the of the stiffness matrix because it operates at the element level. It is evident that the EBE-PCG is advantageous only for large, high bandwidth problems. Even for these problems, though, the effectiveness of the conjugate gradient algorithm for multiple load cases must be examined.
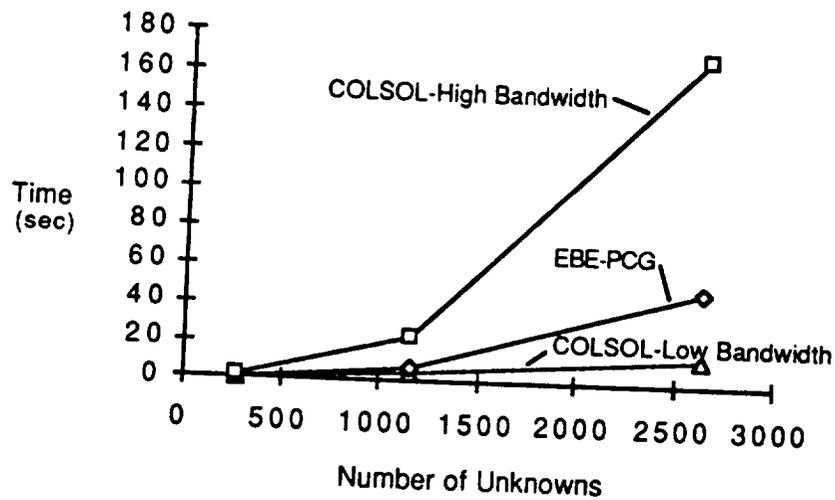
Figure 3. Solution time required for a single load case (20x20 mesh).

Multiple Load Cases

Often, one needs to analyze a system's response to more than one set of loads. For these types of problems, the LDL^T factorization scheme seems to be advantageous, because the factorization of the stiffness matrix only has to be done once.

The iterative process of the conjugate gradient algorithm generates information about the system (namely the [K]-conjugate directions, $\{p\}_i$). It would appear that this information can be saved and used during subsequent load cases to reduce the number of iterations required. (See Figure 4.)

```
          ┌──────────────────┐
          │  First load case │
          └──────────────────┘
                   │
                   ▼
┌──────────────────────────────┐      ┌──────────────────────────────┐
│ Conjugate Gradient Algorithm │◄────►│ Save direction vectors {p}ᵢ  │
└──────────────────────────────┘      └──────────────────────────────┘
                   │
                   ▼
┌──────────────────────────────────────┐
│ Form matrix [P] = [{p}₁,{p}₂,...,{p}ₘ │
└──────────────────────────────────────┘
                   │
                   ▼
┌──────────────────────────────────────┐
│ Let [D] be the diagonal matrix [P]ᵀ[K][P] │
└──────────────────────────────────────┘
                   │
                   ▼
       ┌──────────────────────┐
       │ Subsequent load cases│
       └──────────────────────┘
                   │
                   ▼
┌────────────────────────────────────────────┐
│ Use {q} = [P][D]⁻¹[P]ᵀ{F} as a good first guess │
└────────────────────────────────────────────┘
```
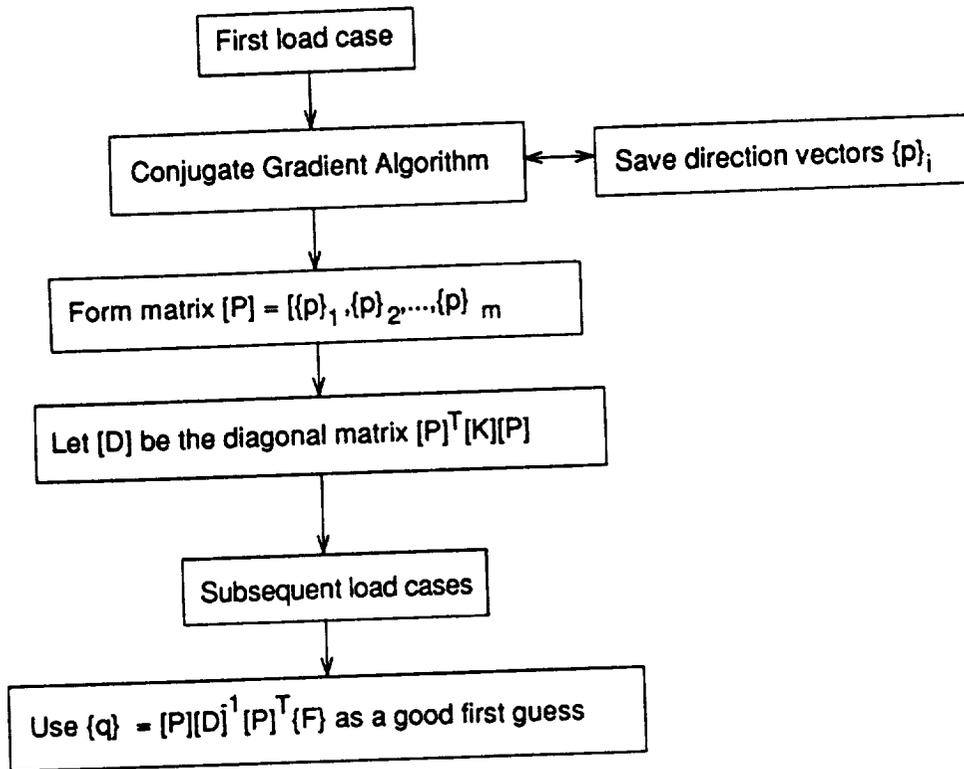
Figure 4. Scheme to improve convergence for multiple load cases.

Given the n x m matrix [P], whose columns are the conjugate directions, $\{p\}_i$, then $[P]^T[K][P] = [D]$ (a diagonal matrix). Note that for all of the conjugate directions (i.e. m = n), $[P][D]^{-1}[P]^T = [K]^{-1}$. Thus, a good initial guess, $\{q\}_0$ is $[P][D]^{-1}[P]^T\{F\}$.

It has been found that the success of using this scheme for multiple load cases is strongly dependent on the the similarity of the load cases. That is, if two load cases both are symmetric with respect to problem geometry, then there is great improvement. However, with two general load cases, only a modest improvement can be expected (See Table II.). This would seem to suggest that in solving nonlinear problems with a modified Newton-Raphson technique with load incrementation, the proposed EBE-PCG may be cost effective, but that remains to be established pending further investigation.

Table II. Results of Multiple Load Case Tests

Load Case (20x20 mesh)	Number of iterations (as the only load case)	Number of iterations (as the second load case using output from the single midpoint load)
Single midpoint load	102	11
Single mid-quarter point load	174	123
Two opposite mid-quarter point loads	102	51 .
Single load 1 node from midpoint	220	179

Figure 5. shows the solution times required for both the EBE-PCG and for COLSOL to solve two successive identical load cases for the same models described previously. Note that this represents the ideal multiple load case problem to the conjugate gradient algorithm. The trend appears very similar to that observed for the single load case. However, as Figure 6. shows, even for a low bandwidth problem, the additional time required for the EBE-PCG to solve a general second load case is always greater than that required by COLSOL. These results indicate that the effectiveness of the conjugate gradient algorithm will be severely limited for large numbers of load cases in linear analysis.
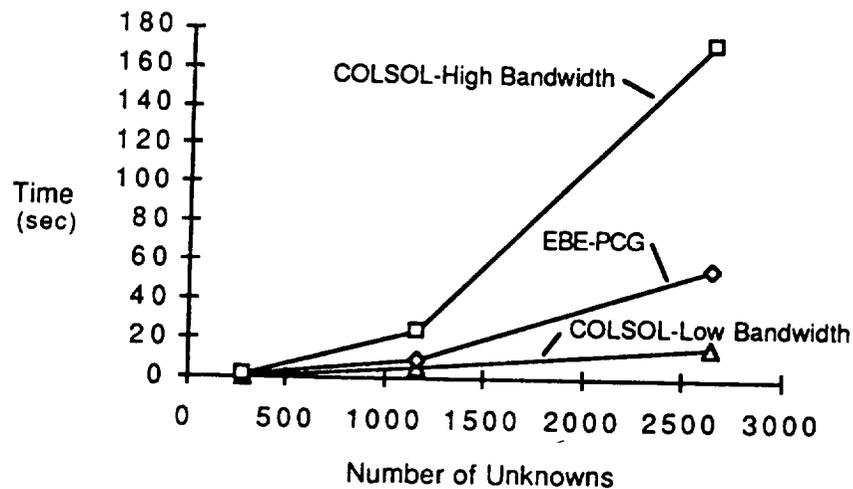
Figure 5. Solution time required for two identical load cases (20x20 mesh).

535
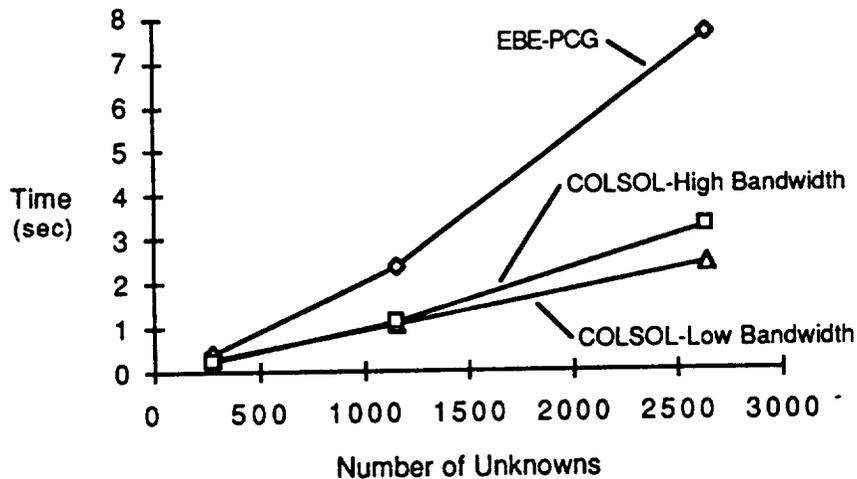
Figure 6. Additional time required for the second load case.

References

[1] Hestenes, M.R. , *Conjugate Direction Methods in Optimization*, pp. 116-125, Springer-Verlag, 1980

[2] Law, K.H. ," Parallel Finite Element Solution Method", *Computer & Structures* vol.23, No.6, pp845-858,1986

[3] Argyris, J. et al., "Τα Παντα Ρει", *Computer Methods in Applied Mechanics and Engineering*, vol.51, pp.289-365,1985

[4] Hughes, T.J.R. , *The Finite Element Method – Linear Static & Dynamic FE Analysis*, pp. 483-489, Prentice-Hall, 1987

[5] Bathe, K.J. and Wilson, E.L., *Numerical Methods in Finite Element Analysis* ,pp. 256-258, Prentice-Hall, 1976

[6] Ayers, C.L. , "Concurrent Processing of Finite Element Calculations", M.S. thesis, Georgia Institute of Technology, July 1989.

[7] FX/Fortran Programmer's Handbook, ALLIANT COMPUTER SYSTEMS Company, v.4.0, 1988

[8] Concentrix System User's Guide, ALLIANT COMPUTER SYSTEMS Company, v.1.0, 1988

STACKING OPTIMIZATION OF COMPRESSOR BLADES OF GAS TURBINE ENGINES

TSU-CHIEN CHEU

TEXTRON LYCOMING

STRATFORD, CONNECTICUT

U.S.A.

ABSTRACT

A procedure is presented to obtain optimal designs of axial compressor blades with structural design constraints. Coefficients of the polynomials defining the circumferential tilting angles and the axial leaning distances of the airfoil cross sections from the initial design geometry are used as design variables. The compressor blades are modeled by 20-node solid elements. An efficient finite element method is developed for modal analysis and sensitivity analysis with respect to the design variables. Based on this information, a sequential linear programming method is applied to calculate the required change of geometry for the desired structural design constraints.

1. INTRODUCTION

In order for a gas turbine engine to achieve its designed performance, the compressor rotors of the engine must be designed to deliver the air at a compression ratio above a certain specified minimum level. However, the compressor blades designed to achieve the required aerodynamic performance often cannot meet the structural design criteria, e.g. stress level and natural frequency distribution. Under this condition, blades have to be designed iteratively to satisfy both aerodynamic and structural design specification, [1-7]. In order to obtain an optimal blade design with minimum expenditure of design effort, appropriate design parameters and guide lines must be established for this iterative procedure.

A blade design usually begins with an aerodynamically optimized shape, [8,9]. Since the blade height, chord length and each cross sectional airfoil shape and its orientation with respect to the axis of rotation have primary influence on the aerodynamic performance of the blade, it is not appropriate to change these geometric characteristics of a blade having an aerodynamically optimized shape. Therefore, the design parameters permitted to vary are the circumferential tilting angles and the axial leaning distances of the airfoil cross sections with respect to their locations in the initial design. Definitions of these geometry parameters are shown in Figure 1.

Because moderate alterations of the airfoil cross sectional tilting angles and leaning distances have only secondary effect on the blade aerodynamic performance. Design iterations between aerodynamics and structural constraints can be easier achieved using these parameters as design variables of the blade design optimization.

Since the circumferential tilting and axial leaning of the airfoil cross sections from their initial design locations modify the way these cross sections are stacked to form a blade, this method is called blade stacking optimization.

538
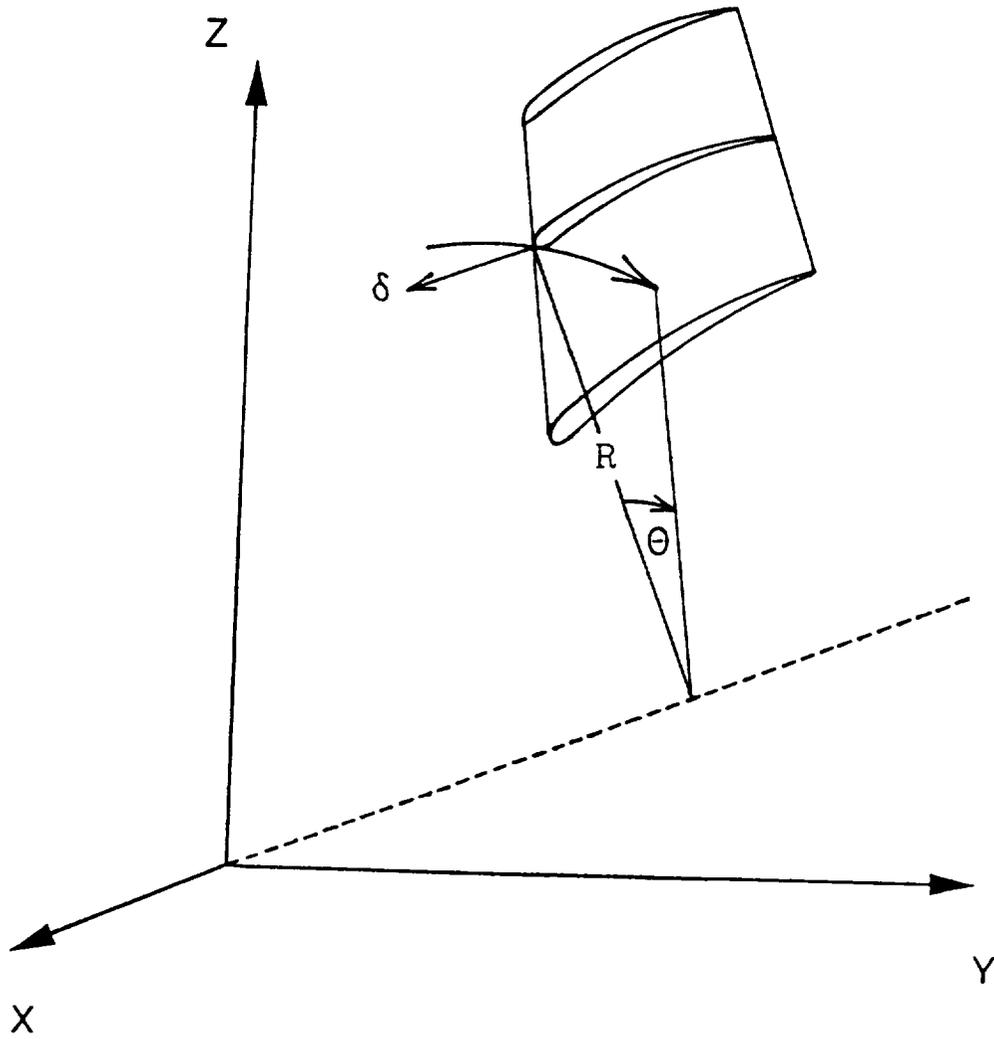
Figure 1. Definitions of Blade Airfoil Cross Sectional Design Variables
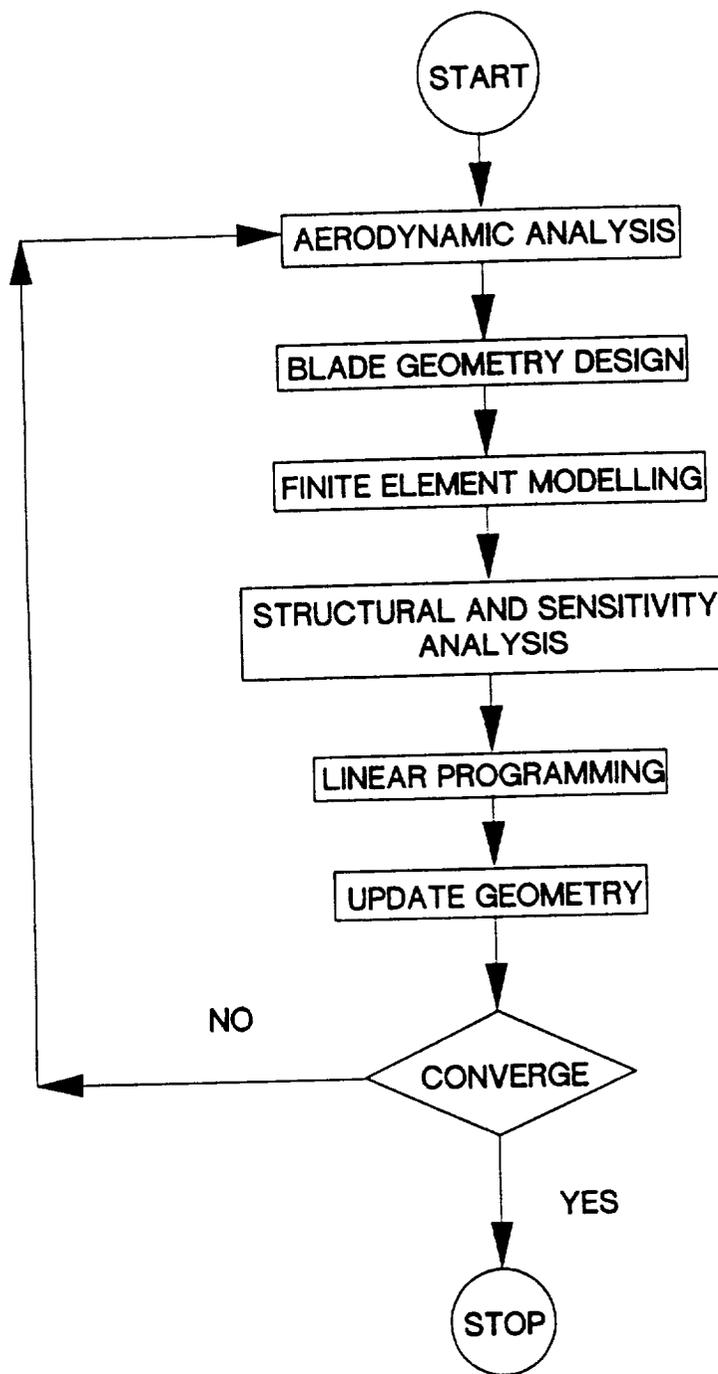
Figure 2. Overview of Blade Design Optimization

In this paper a procedure of blade stacking optimization is presented. An overview of the procedure is shown in Figure 2.

2. FORMULATIONS

Using cubic polynomials to fit the curve passing through the stacking points of a blade, the circumferential tilting angle and the axial leaning distance of the airfoil cross sections can be expressed as functions of radial distances of the stacking points from the axis of rotation

$$\theta = a_1 + a_2 r + a_3 r^2 + a_4 r^3$$

$$X = a_5 + a_6 r + a_7 r^2 + a_8 r^3$$

The problem can be stated as to minimize the global change of the polynomial coefficients

$$F = \sum_{m=1}^{NV} (a_m - a_{mo})^2$$

subject to the frequency constraints on the jth mode

$$f_j^l \le f_j \le f_j^u \qquad\qquad j = 1, \ldots, NM$$

or their corresponding eigenvalue constraints

$$\lambda_j^l \le \lambda_j \le \lambda_j^u \qquad\qquad j = 1, \ldots, NM$$

with side constraints on the polynomial coefficients.

$$a_m^l \le a_m \le a_m^u \qquad\qquad m = 1, \ldots, NV$$

where NM is the total number of natural frequency constraints and NV is the number of design variables.

Using the first order Taylor series expansion F and λ of the current design can be approximated as

$$F - F_o + \sum_{m-1}^{NV} \frac{\partial F}{\partial a_m} (a_m - a_{mo})$$

$$\lambda_j - \lambda_{jo} + \sum_{m-1}^{NV} \frac{\partial \lambda_j}{\partial a_m} (a_m - a_{mo})$$

The derivatives of the objective function can be easily found as

$$\frac{\partial F}{\partial a_m} - 2 a_m$$

whereas much more computation efforts are required to find the derivatives of the eigenvalues.

The stiffness and mass matrices of the 3-dimensional solid elements in the global Cartesian coordinates are

$$K - \iiint B^T D B \, dx \, dy \, dz$$

$$M - \iiint \rho N^T N \, dx \, dy \, dz$$

or in the element natural coordinates as

$$K - \iiint B^T D B |J| \, d\xi \, d\eta \, d\zeta$$

$$M - \iiint \rho N^T N |J| \, d\xi \, d\eta \, d\zeta$$

where N is the shape function, B is the displacement differential operator, D is the elastic material matrix, ρ is mass density and $|J|$ is the determinant of the Jacobian transformation matrix, J, between the Cartesian and natural coordinate systems.

The free vibration equation for the finite element model is

$$K \phi_j - \lambda_j M \phi_j$$

542

with the eigenvectors, ϕ_j, normalized to satisfy the following conditions

$$\phi_j^T M \phi_j - I$$

$$\phi_j^T K \phi_j - \lambda_j$$

Taking the derivatives of the stiffness and mass matrices as

$$\frac{\partial K}{\partial a_m} - \iiint \left(\frac{\partial B^T}{\partial a_m} DB|J| + B^T D \frac{\partial B}{\partial a_m} |J| + B^T DB \frac{\partial |J|}{\partial a_m} \right) d\xi \, d\eta \, d\zeta$$

$$\frac{\partial M}{\partial a_m} - \iiint \rho N^T N \frac{\partial |J|}{\partial a_m} d\xi \, d\eta \, d\zeta$$

and using the normalized eigenvectors, the derivatives of the eigenvalues can be calculated from

$$\frac{\partial \lambda_j}{\partial a_m} - \phi_j^T \left(\frac{\partial K}{\partial a_m} - \lambda_j \frac{\partial M}{\partial a_m} \right) \phi_j$$

as shown in Ref [10].

With these derivatives, the objective function and constraints can be linearized to form a linear programming problem, [11,12]. After the blade geometry is modified for the current iteration, the new geometry becomes the initial design for the next iteration step. The computation procedure is repeated until the desired structural design constraints are satisfied.

REFERENCES

1. DeSilva, B.M.E., Negus, B. and Worster, B., "Penalty Function Type Optimal Control Methods for the Design of Turbine Blade Profiles", NASA Report N76 11464, 1975.

2. DeSilva, B.M.E., Grant, G.N.C. and Pierson, B.L.,"Conjugate Gradient Optimal Control Methods for the Design of Turbine Blades", NASA Report N76 11463, 1975.

3. DeSilva, B.M.E. and Grant, G.N.C., "Comparison of Some Optimal Control Method for the Design of Turbine Blades", ASME Paper No. 77-Det-43, 1978.

4. Queau, J.P. and Trompette, P., "Optimal Shape Design of Turbine Blades", ASME J. of Vibration, Acoustics, Stress, and Reliability in Design", Vol. 105, pp. 444-448, 1983.

5. Stetson, K.A. and Harrison, I.R., "Redesign of Structural Vibration modes by Finite-Element Inverse Perturbation", ASME J. of Eng. for Power, Vol. 103, pp. 319-325, 1981.

6. Cheu, T.C., Wang, B.P. and Chen, T.Y., "Design optimization of Gas Turbine Blades with Geometry and Natural Frequency Constraints", Computers & Structures, Vol. 32, pp. 113-117, 1987.

7. Brown, K.W., "Aero/Structural Tailoring of Engine Blades (Aero/STAEBL)", NASA Report CR-180805, 1988.

8. Law, C.H. and Puterbough, S.L., "A Computer Program for Axial Compressor Design (UD0300M)", AFWAL-TR-82-2074, 1982.

9. Chang, J.F. and Lan, C.E., "Transonic Airfoil Analysis and Design in Nonuniform Flow", NASA Report CR-3991, 1986.

10. Fox, R.L. and Kapoor, M.P., "Rate of Change of Eigenvalues and Eigenvectors", AIAA Journal, Vol. 6, No. 12, pp.337-351, 1968.

11. Zienkiewicz, O.C. and Campbell, J.S., "Shape Optimization and Sequential Linear Programming", Chapter 7 of Optimum Structural Design Theory and Applications, Edited by Gallagher, R.H. and Zienkiewciz, O.C., John Wiley and Sons, 1973.

12. Vanderplaats, G.N., Numerical Optimization Techniques for Engineering Design: With Applications, McGraw-Hill, 1984.

Todd J. Beltracchi
Member of Technical Staff
Optimization Techniques Section
The Aerospace Corporation
P.O. Box 92957
Los Angeles, California 90009-2957

ABSTRACT

The high cost of delivering payloads to orbit demands that launch vehicles be flown in an optimal and reliable manner. Thus, computer simulated trajectories are optimized to obtain maximum vehicle payload capability or a maximum amount of reserve fuel, subject to design constraints.

The Generalized Trajectory Simulation (GTS) system developed at The Aerospace Corporation is capable of simulating and optimizing trajectories for a wide variety of boost and reentry problems. Optimization is an integral part of the GTS system. This has resulted in progressive use of optimization. Additionally, the flexibility of the GTS system allows optimum trade studies to be conducted and complicated missions to be designed.

This paper describes the integrated and flexible nature of the optimization software in the GTS system. Special features of the NLP2 and NLP3 optimization codes are described. Post optimality, parameter sensitivity analysis and the automatic scaling capability in the GTS system are also described.

INTRODUCTION

The high cost of vehicle simulation and modular design of the GTS system has led to the development of two powerful optimization codes, NLP2[3] (A Generalized Projected Gradient (GPG) code) and NLP3[20] (A Generalized Reduced Gradient (GRG) code). Finite difference approximations are needed to calculate the gradients of the functions. GTS contains algorithms that automatically choose the best perturbation sizes for the partial derivatives. Additionally, POSTOP[15], a post optimality / parameter sensitivity analysis[18] / automatic scaling[16,17] capability has been developed as part of GTS.

This paper begins with a definition of the standard form of the nonlinear programming problem. Next, trajectory optimization in the GTS system is described. A sample problem is provided to describe the simulation of the vehicle and optimization problem set up. The next section describes and compares the optimization algorithms. The final section describes the post optimality analysis capability.

STANDARD FORM

The standard form of the nonlinear programming problem (NLP) is

Minimize: $f(x,p)$
Subject to:

$$c_i(x,p) = 0 \quad i=1,NE \tag{1}$$
$$c_i(x,p) \geq 0 \quad i=NE+1,m \tag{2}$$
$$x = (x_1, x_2, \ldots, x_n) \tag{3}$$
$$p = (p_1, p_2, \ldots, p_k) \tag{4}$$
$$\tag{5}$$

Where f represents the objective function, c represents the constraints, NE is the number of equality constraints, m is the total number of constraints $(m=NE+NI)$, and NI is the number of inequality constraints. The design variables, x, are chosen by the optimization operator. The design parameters, p, are nominally considered constant but may vary due to uncertainties in modeling or user re-specification.

The first order Kuhn-Tucker conditions [13,23] are checked to verify optimality.

$$\nabla_x L = 0$$
$$\lambda_i c_i(x,p) = 0 \quad i=1,m$$
$$c_i(x,p) = 0 \quad i=1,NE$$
$$c_i(x,p) \geq 0 \quad i=NE+1,m$$
$$\lambda_i \geq 0 \quad i=NE+1,m \tag{6}$$

with the Lagrangian function

$$L(x,p) = f(x,p) + \sum_{i=1}^{m} \lambda_i c_i(x,p) \tag{7}$$

where λ_i are the Lagrange multipliers. The second order Kuhn-Tucker optimality conditions are

$$y^T \nabla_x^2 L y > 0 \tag{8}$$

for any y such that

$$y^T \nabla_x c_A = 0. \tag{9}$$

where the active set of constraints c_A contains all equality constraints and any inequality constraints satisfied at their bounds. The number of constraints in the active set (m_A) is less than or equal to the number of variables.

The number of degrees of freedom in the optimization problem is equal to the number of design variables minus the number of constraints in the active set. Problems with few degrees of freedom are generally easier to solve than problems with many degrees of freedom.

GENERALIZED TRAJECTORY SYSTEM

GTS [5,6,9 and 22] was designed to simulate and optimize trajectories, and has undergone continuous development for the past 25 years. GTS allows for trajectory simulation of a wide variety of boosters, upper stages, satellites, and reentry vehicles. The trajectory is broken into events and phases over which the equations of motion are integrated. Typical events can correspond to vehicle events such as liftoff, staging, maximum dynamic pressure, etc.

Any simulation input can be treated as an optimization variable, and any output can be constrained or used as the objective function. Typical objective functions are payload, reserve fuel, reentry cross range, or delta V. Equality constraints are often used to restrict the final state vector to be in the mission orbit, or at the desired location. Inequality constraints may be placed on vehicle attitude, velocity, or heating at staging, or dynamic pressure. Design variables can be pitch rates for the stages, burn times, coast times, time of payload fairing jettison, payload weight, etc. The design parameters are typically the fuel loaded, inert weight of the stages, or launch conditions (e.g., temperature or winds).

Trajectory optimization problems in GTS normally have fewer than 25 design variables, with 1 to 10 degrees of freedom. The problems are poorly scaled and have very nonlinear constraints. Trajectories are usually simulated by numerically integrating the equations of motion. The simulations are noisy, making them difficult to optimize.

The vehicles can be simulated in three degree of freedom (3 DOF) or 6 DOF. Optimization is generally only applied to the 3 DOF trajectory simulations.

The basic equations of motion are integrated

$$\Sigma F = m\, a$$

The forces can be gravity, aerodynamic, solar radiation pressure, and thrust. There is also mass flow i.e., ablation, propellent, or inert weight flow. The forces are generally functions of time and initial conditions. The equations of motion are numerically integrated to obtain a 3 DOF approximation of the trajectory. Several integration methods[5] (Runge Kutta, Adams, etc.) are available. The user can specify the step size and order used by the integration. The integration is restarted at each event, because events such as staging cause discontinuities in the equations of motion.

Vehicle definitions are created by combining engine, weight flow, propulsion, control, aerodynamic, gravity, etc., models into a sequence of events. The equations of motion are integrated over phases between the events. The user can specify auxiliary functions defined in FORTRAN that perform computations not built into GTS. Data can be input in a convenient tabular format, or through user defined subroutines.

Several operators (e.g., TRAJCEM, DERSYS, OPTSYS, SIZE[6], MCARLO[22] and PSTUDY) are available in GTS. TRAJCEM, DERSYS, and OPTSYS are described in this paper, which focuses on the OPTSYS operator.

The TRAJCEM operator executes a series of commands that simulates a single trajectory. Auxiliary computations can be performed at any event or every integration point.

The DERSYS operator is used to evaluate derivatives of any output quantity with respect to any input quantity. DERSYS can use TRAJCEM as a function generator. Forward or central difference approximations can be used. Techniques have been developed for choosing appropriate values of the perturbation sizes for calculating the partials[19].

The OPTSYS operator executes the specified optimization algorithm for a problem definition. OPTSYS can use TRAJCEM as a function generator. The OPTSYS operator can also be used for constraint solving problems, i.e. the user desires a feasible trajectory. When an optimization algorithm requests a gradient vector, finite difference approximations are calculated using the same techniques as DERSYS. Error control for gradient approximations can be executed once or every time a gradient evaluation is requested by the optimization algorithm.

GTS uses Generalized Trajectory Language (GTL) to define problems. A sample GTL input description for a two-stage booster (Figure 1) is provided. Some of the GTL input is omitted for brevity. Figure 2, a schematic of the event sequence, illustrates the pitch rates during the trajectory.
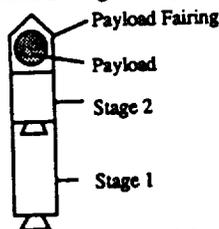
Figure 1: Generic Two-Stage Vehicle

The vehicle is initialized at event P000. Ignition occurs at event P001, 5 seconds after ignition (event P005) the first inertial pitch rate QR1 begins, and it is held for 5 seconds. At event P006 a gravity turn is flown for 60 seconds until event P010. Some time between events P006 and P010 the maximum dynamic pressure (event PUQMAX) occurs. The second pitch rate, QR2, is used from event P010 to P019. Event P019 occurs when the propellent tank of the first stage is empty, at this point stage one is jettisoned, and stage two ignition occurs 2 seconds later at event P020. The third pitch rate, QR3, is used from event P020 to P025. The fourth pitch rate, QR4, is used from event P025 to P040. The payload fairing is

jettisoned at event P030, where the value of the free molecular heating (FMH) is saved into a variable FMHPLFJ. Finally, at event P040 the state vector is saved.
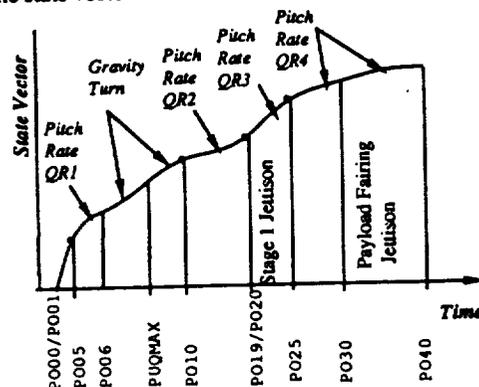
Figure 2: Event Sequence for a Two-Stage Vehicle

```
P000
    PH00 INITIALIZE VEHICLE
        Choose earth, control, engine, aerodynamic
        weight and integration models for stage one

P001 .NULL.
    PH01 IGNITION
        ENGINE1 ENGM1    STG   = 1
                         ENG   = 1
                         FVAC  = 150000.0
                         DWVAC = 5000.0

P005 P001(TFROM) = 5.0
    PH05 BEGIN PITCH RATE 1
        CONTROL CNTRLM6  QC = QR1

P006 P005(TFROM) = 5.0
    PH06 FLY GRAVITY TURN
        CONTROL CNTRLM1

PUQMAX QDYNP .MAX.

P010 P006(TFROM) = 60.0
    PH10 BEGIN PITCH RATE 2
        CONTROL CNTRLM6  QC = QR2

P019 WTANK(1,1) = 0.0
    PH19 JETTISON STAGE 1
        Drop stage 1 models

P020 P019(TFROM) = 2.0
    PH20 SECOND STAGE INITILIZATION PITCH RATE 3
        Choose models for stage 2
        ENGINE2 ENGM1    STG   = 2
                         ENG   = 1
                         FVAC  = 140000.0
                         DWVAC = 4000.0
        CONTROL CNTRLM6  QC = QR3

P025 P020(TFROM) = 10.0
    PH10 BEGIN PITCH RATE 4
        CONTROL CNTRLM6  QC = QR4

P030 TIME = TPLFJ
    PH30 JETTISON PAYLOAD FAIRING
    /SAVE FMHPLFJ = FMH/

P040 WTANK(2,1) = 0.0
    /SAVE RVFIN  = RV
         VIFIN  = VI
         GAMFIN = GAMMAI/
    /STOP/
```

GTS can exploit the problem structure to reduce the time required for evaluation of partial derivatives by using a mechanism called partial trajectories. In the example problem, QR4 does not affect the trajectory until event P025. Thus, when

$$\frac{\partial f}{\partial QR4} = \frac{f(QR4+\Delta QR4) - f(QR4)}{\Delta QR4}$$

is evaluated, the equations of motion are integrated from event P025 to P040 (initialized using the state vector at event P025) to obtain $f(QR4+\Delta QR4)$. The user specifies the event where the

546

variable begins to affect the trajectory, so the system knows where to save the state vectors and where to begin integration of the equations of motion when partial derivatives are evaluated. This reduces the cost of gradient evaluation by reducing the cost of evaluating perturbed trajectories.

Optimization Problem Input

An input file for GTS consists of a trajectory definition, input data tables (describing atmosphere, propulsion, etc.), FORTRAN subroutines (used for auxiliary computations), OPTSYS or DERSYS problem definitions, and input data. A sample optimization problem definition for the generic two stage vehicle is:

```
BOOSTER          BOOSTER OPTIMIZATION
    OPTALG  NLP2
            PROB=:MAX:
            :
            algorithm parameters
            :
            /THE FUNCTION GENERATOR IS TRAJCEM/
    OBJFTN  OBJFNM1    (PAYLOAD)
            OBJSKL = 1.E-3
            OBJTOL = 1.E-4
            PGDTOL = 1.E-4
    CONSTR  CNSTRM1
            CONMAXQ CONFMH
            CONRV   CONVI  CONGAMMA
    INDVAR  VARM1
            VPITCH1 VPITCH2 VPITCH3 VPITCH4
            VTPLFJ  VPAYLOAD
    OPTAOM  OPTOM1
            (HFT.AT.PO30)
```

BOOSTER defines and names the problem, and OPTALG is used to select the optimization algorithm and specify the algorithm parameters. If the algorithm parameters are not provided, default values are used. The OBJFTN, CONSTR, INDVAR, and OPTAOM inputs are explained in the following paragraphs.

The objective function is specified under OBJFTN. OBJSKL is the scale factor for the objective function. OBJTOL and PGDTOL are convergence parameters for the objective function and projected gradient.

Constraint selection is made under CONSTR. Constraints are defined as

```
CONMAXQ    (QDYNP .AT. PUQMAX ≤ QMAX)
           CONTOL = 1.E-6
           CONSKL = 1.E+2
           INBASIS = :YES:
```

This constraint is used to keep the maximum dynamic pressure, (QDYNP at event PUQMAX), less than or equal to the limit QMAX. The tolerance for each constraint is specified as CONTOL. The constraint scaling is specified as CONSKL. The INBASIS option is used to identify an inequality constraint as a potentially active constraint for the optimization.

The variable names are entered under INDVAR. A variable (or parameter) definition is

```
VPITCH4    VAR = (QR4]
           LWRBND = -0.1
           UPRBND = 0.1
           STARTAT = [PO25]
           PDTYPE = :2-SIDED:
           DELVAR = 1.E-4
           VARSKL = 1.E+2
           ABSTOL = 1.E-5
           CSOLVE = :YES:
```

Here VPITCH4 is a pitch rate (QR4) input to the trajectory. There are lower and upper bounds assigned. The bounds define computability regions. If the optimization attempts to evaluate the trajectory outside of these bounds a function error is returned. STARTAT is used by the partial trajectories option to set an integration restart point at event PO25. The partial derivative type (PDTYPE) specification of :2-SIDED: is for central difference

approximation for partial derivatives, where DELVAR is the initial perturbation size to be used when evaluating the partials. VARSKL is the scale factor for the variable. The ABSTOL parameter is used to specify the convergence tolerance for the variable. The CSOLVE option can be used to specify the constraint solving variables for the NLP3 code.

OPTAOM is used to specify quantities that are included in the iteration output during the optimization. These are items that the user may want to monitor, such as the altitude at payload fairing jettison.

GTS has a restart capability that is useful for solving optimization problems requiring a large amount of CPU time. When the run is close to the time limit, GTS saves the information required for a restart into a file. The file is used as input for the next run and the optimization algorithm is restarted as if there were no interruption. Any Hessian approximation or other information that the code had created is available for the subsequent run.

OPTIMIZATION ALGORITHMS

NLP2 and NLP3 are optimization programs developed at The Aerospace Corporation to solve the nonlinear programming problems generated in GTS. Some common features of the codes are described first, followed by a discussion of the algorithms, the nonlinear least squares capability, and a comparison of the algorithms.

Common Features

Both codes are feasible point methods, thus they produce a series of feasible points approaching the solution. If the starting point is infeasible, the codes use built in searches to locate the constraint surface. A secant method with Broyden updates to the constraint Jacobian that is used during the line search. The secant update improves the performance of the constraint solving algorithm and reduces the number of function evaluations (due to constraint solving) required to solve the problem.

The codes use an active set method to solve a series of equality constrained subproblems until the solution is reached. Each subproblem is terminated when either a local minimum is reached or an inequality constraint must be added to the active set. When the subproblem is solved, a check of the Lagrange multipliers for the inequality constraints is made to verify that the correct active set of inequality constraints has been obtained. Some implementations of the GRG method have used slack variables to deal with inequality constraints[11,12,23]. Using the active set method for the inequality constraints reduces the size of the matrices that must be manipulated, although the complexity of the code is increased due to the additional code required to determine the active set.

A BFS/SR1 variable metric update is used to approximate the restricted Hessian (H_R) and accelerate the search for the minimum. The restricted Hessian is defined as the projected Hessian (for NLP2 a GPG code) or reduced Hessian (for NLP3 a GRG code). The search direction is calculated using

$$s = H_R^{-1} \frac{df_R}{dx}$$

(10)

Where df_R/dx is the restricted gradient. The BFS[13] update is used as the primary update to construct the approximation of the Hessian, if

$$\Delta x^T \Delta \frac{df_R}{dx} < 0$$

(11)

a switch is made to the Symmetric Rank One (SR1)[13] update, because Eq. (11) violates the convexity assumption used by the BFS update. The SR1 update is then used until the problem is solved or there is a change in the active set. The initial Hessian approximation is taken as the identity matrix. The Hessian approximation is reset when there is a change in the active set.

The Lagrange multipliers are calculated from the least squares estimate

$$\min_{\lambda} \| \frac{df^T}{dx} - \lambda^T \frac{dc_A}{dx}^T \|_2 \qquad (12)$$

using a QR factorization of the active constraints.

Several different convergence criteria tests are used to determine optimality. Single tests are easily fooled by the complicated and poorly scaled characteristics of trajectory optimization problems. The codes check for convergence of the constraints to within their tolerances, the correct set of inequality constraints has been obtained, and either the objective function has converged or the projected/reduced gradient is within the specified tolerance. Additionally there are checks on small changes in the design variables.

Problem scaling is important for successful solution of problems[1,13,16,23]. The codes solve scaled optimization problems, however unscaled results are reported to the users. The scaling transformations are

$$f_s = \alpha f \qquad (13)$$
$$c_s = E c \qquad (14)$$
$$x_s = D x \qquad (15)$$
$$p_s = B p \qquad (16)$$

where α is a scalar and E, D, and B scaling matrices are of the form

$$E = Diag(e_1, e_2, ..., e_m)$$
$$D = Diag(d_1, d_2, ..., d_n)$$
$$B = Diag(b_1, b_2, ..., b_k)$$

The scaling transforms the function gradients,

$$\nabla_{x_s} f_s = \alpha D^{-1} \nabla_x f \qquad (17)$$

$$\nabla_{x_s} c_s = E \nabla_x c D^{-1} \qquad (18)$$

and the Hessian of the Lagrangian,

$$\nabla_{x_s}^2 L_s = \alpha D^{-1} \nabla_x^2 L D^{-1} \qquad (19)$$

The Lagrange multipliers for the unscaled problem are calculated from

$$\lambda = \alpha^{-1} E \lambda_s \qquad (20)$$

The codes require the input of the scale factors.

If the functions cannot be evaluated at a point requested by the algorithm, a function error flag is returned. The algorithms can recover from function errors when the initial point is computable. Function errors can occur in GTS when the trajectory cannot be propagated to completion. When a large perturbation of the variables occurs in the line search it can cause a function error and the step length is reduced until a new computable point is found. After several iterations the procedure to select the initial step length in the line search and the convergence of the Hessian approximation lead to smaller changes in the design variables and no longer produce function errors. Users specify a limit on the number of successive function errors allowed before the algorithm quits.

The codes can be run in a stand alone manner and they assume the objective function and constraints are evaluated as a set, i.e., from some auxiliary black box function generator. A reverse communication architecture is used in the interface.

NLP2: A Generalized Projected Gradient Algorithm

The NLP2[3] code has been developed over the past 10 years. NLP2 is a generalized projected gradient (GPG) algorithm. NLP2 has over 20 input parameters which allows a great deal of control over the inner workings of the code. Experienced users can tune NLP2's parameters for rapid solution for particular formulations.

NLP2 uses a QR orthogonal factorization of the active constraint set

$$\frac{dc_A}{dx} = QR \qquad (21)$$

Q is orthogonal, and R is upper triangular. The Q and R matrices are partitioned as

$$Q = [Q_1 | Q_2] \quad R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$$

where the columns of Q_1 form a basis of the constraint Jacobian

$$Q_1^T \frac{dc_A}{dx} = R_1 \qquad (22)$$

and the columns of Q_2 form a basis for the null space of the constraint Jacobian

$$Q_2^T \frac{dc_A}{dx} = 0 \qquad (23)$$

The matrix Q can be used to transform the coordinate axes x to y.

$$y = Q^T x \qquad (24)$$
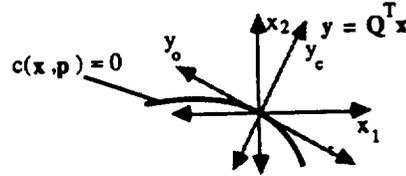
Figure 3: Transformed Coordinate Axes

NLP2 solves the problem as a generalized reduced gradient problem in the y variables. The y variables are divided into optimization (y_o in Figure 3) and constraint solving (y_c in Figure 3) variables

$$y = [y_c, y_o] \qquad (25)$$

The optimization variables (y_o) are locally tangent to the constraint surface and the constraint solving variables (y_c) are locally normal to the constraint surface. The reduced gradient in the y variables is defined by differentiating the objective function with respect to y_o with the y_c variables adjusted to maintain constraint feasibility to obtain

$$\frac{df}{dy_o} = \frac{\partial f}{\partial y_o} + \frac{\partial y_c}{\partial y_o} \frac{\partial f}{\partial y_c} \qquad (26)$$

In terms of the x variables this equation is

$$\frac{df}{dy_o} = \left[Q_2^T + \frac{\partial y_c}{\partial y_o} Q_1^T \right] \frac{df}{dx} \qquad (27)$$

the derivatives of the active constraints with respect to the optimization variables are 0, thus

$$\frac{dc_A}{dy_o} = \left[Q_2^T + \frac{\partial y_c}{\partial y_o} Q_1^T \right] \frac{dc_A}{dx} = 0 \qquad (28)$$

which can be solved for

$$\frac{\partial y_c}{\partial y_o} = - Q_2^T \frac{dc_A}{dx} [Q_1^T \frac{dc_A}{dx}]^{-1} \qquad (29)$$

This is substituted into equation 27 to obtain

$$\frac{df}{dy_o} = \left[Q_2^T - Q_2^T \frac{dc_A}{dx} [Q_1^T \frac{dc_A}{dx}]^{-1} Q_1^T \right] \frac{df}{dx} \qquad (30)$$

using the definition in equation (23), yields

$$\frac{df}{dy_o} = Q_2^T \frac{df}{dx} \qquad (31)$$

The search direction for variables in the y space is

$$s_y = \begin{bmatrix} s_{y_c} \\ s_{y_o} \end{bmatrix} = \begin{bmatrix} s_o \frac{\partial y_c}{\partial y_o} \\ H_p^{-1} \frac{df}{dy_o} \end{bmatrix} \qquad (32)$$

where H_p is an approximation of the projected Hessian (initially taken as identity matrix). $\partial y_c / \partial y_o$ is equal to zero when $Q_2 dc_A /dx = 0$, thus

$$s_y = \begin{bmatrix} 0 \\ H_p^{-1} \frac{df}{dy_o} \end{bmatrix} = \begin{bmatrix} 0 \\ H_p^{-1} Q_2^T \nabla_x f \end{bmatrix} \qquad (33)$$

The search direction is transformed into the x variables using

$$x = Q y \qquad (34)$$

Thus the search direction in the x variables is

$$s_x = Q s_y = Q \begin{bmatrix} 0 \\ H_p^{-1} Q_2^T \nabla_x f \end{bmatrix} \qquad (35)$$

which is equivalent to the projected gradient in the x variables. The search direction obtained from the traditional projected gradient

$$\nabla_p f = \left(I - \frac{dc_A}{dx}^T \left(\frac{dc_A}{dx} \frac{dc_A}{dx}^T \right)^{-1} \frac{dc_A}{dx} \right) \frac{df}{dx} \quad (36)$$

is numerically unstable[14], thus equation (35) is recommended.

The Lagrangian function [eq. (7)] is used as a merit function in the line search. Using this merit function removes some of the sensitivity to satisfying the constraints exactly. The Lagrange multiplier estimates are calculated at the initial point in the line search using a least squares estimate [Eq. (12)].

The search for the optimum in NLP2 is accelerated using the BFS/SR1 update to create an approximation of the projected Hessian using Δy_0 and $\Delta df/dy_0$. The search direction for the constraint solving variables is calculated from equation (32).

NLP2 has an option that calculates the projected gradient by perturbing the optimization variables (y_0), and adjusting the constraint solving variables to return to the constraint surface.

$$\frac{df}{dy_0} = \frac{f_c(y + \Delta y_0, p) - f(y, p)}{\Delta y_0} \quad (37)$$

where f_c is a feasible point. (This technique has been used for calculation of reduced gradients[23]). NLP2 also has an option to numerically calculate the projected Hessian and then solve the problem using second order information. These techniques are efficient in some applications where there are few degrees of freedom.

The projected Hessian can be discontinuous as the solution is approached due to discontinuities in the Q matrix[4]. This can lead to slow convergence for the GPG method; however, in our experience this has not been observed to cause any problems. NLP2 has an option that allows the tranformation matrix to be fixed on the first iteration and not recalculated on subsequent iterations, thus $Q_2^T dc_A /dx \neq 0$. When this option is used, equations (29), (30) and (32) are used to calculate the projected gradient and search direction, and the restricted Hessian is well behaved approaching the solution.

When the transformation matrix is held constant NLP2 is well behaved when started close to the solution because $Q_2^T dc_A /dx$ remains small. For nonlinear problems that are started far from the solution a new transformation matrix is often required for good performance of the algorithm. NLP2 has built in options for determining when it needs to calculate a new transformation matrix.

NLP3: A Generalized Reduced Gradient Algorithm

The NLP3[20] code has evolved over 20 years and is an implementation of the GRG method[23]. The GRG method divides the design variables into optimization variables x_0 and constraint solving variables x_c

$$x = (x_0, x_c) \quad (38)$$

The constraint solving variables are used to maintain constraint feasibility. The reduced gradient is defined by applying the implicit function theorem to the constraints and examining a perturbation about x yielding

$$\frac{dc_A}{dx}^T \Delta x = C \Delta x_0 + J \Delta x_c = 0 \quad (39)$$

where

$$C = \frac{dc_A}{dx_0}^T \; ; \; J = \frac{dc_A}{dx_c}^T$$

The J matrix is square, and when x_c are chosen such that J is non-singular

$$\Delta x_c = -J^{-1} C \Delta x_0 \quad (40)$$

This can be used to obtain the reduced gradient

$$\frac{df}{dx_0} = \frac{\partial f}{\partial x_0} - J^{-1} C \frac{\partial f}{\partial x_c} \quad (41)$$

This is the partial of the objective function with respect to the optimization variables with the constraint solving variables adjusted to maintain constraint feasibility. When the constraint solving variables are suitably chosen, the reduced gradient is zero at the solution of the problem.

The search direction for the GRG method is composed of a search direction for the optimization variables and a search direction for the constraint solving variables. The search direction is

$$s = \begin{bmatrix} s_{x_0} \\ s_{x_c} \end{bmatrix} = \begin{bmatrix} H_r^{-1} \frac{df}{dx_0} \\ -J^{-1} C \, s_{x_0} \end{bmatrix} \quad (42)$$

where H_r is an approximation of the reduced Hessian. The search for the minimum is accelerated using the BFS/SR1 update to approximate the reduced Hessian using Δx_0 and $\Delta df/dx_0$ as inputs.

NLP3 allows users to specify variables as constraint solving or optimization variables. This is important for problems where the objective function is to minimize or maximize one of the variables. In this case the gradient of the objective function is

$$\frac{df}{dx} = (1,0,0,...,0)$$

If x_1 is an optimization variable the reduced gradient will not converge. The reduced gradient for this problem is

$$\frac{\partial f}{\partial x_0} = (1,0,0,...,0) \; : \; \frac{\partial f}{\partial x_c} = (0,0,...,0)$$

$$\frac{df}{dx_0} = (1,0,0,...,0) - J^{-1} C \, (0,0,...,0)$$

$$\frac{df}{dx_0} = (1,0,0,...,0)$$

which will not go to zero, thus x_1 must be a constraint solving variable.

NLP3 uses a special technique for choosing the constraint solving variables. The method minimizes the l_1 norm estimate of the Lagrange multipliers

$$\min_{\lambda} \| \frac{df}{dx}^T - \lambda^T \frac{dc_A}{dx}^T \|_1 \quad (43)$$

to obtain a set of constraint solving variables. Using standard techniques an equivalent linear programming problem is solved, which is used to define a partition as in equation (38). A complete discussion of this method is available in the NLP3 documentation[20]. This method finds a factorization with a non-singular J unless the problem is singular, and allows for a completely new set of optimization variables to be used when the active set changes.

NLP3 does not explicitly calculate J^{-1} but calculates a QR factorization of J. The QR factorization provides an estimate of the condition number of the active constraint Jacobian.

The user can specify a Newton method be used instead of the secant method to return to the constraint surface in the line search.

Nonlinear Least Squares Capability

Problems in trajectory reconstruction, curve fitting[21], constraint solving or targeting can be formulated as nonlinear least squares problems.

NLP2 and NLP3 have a built-in capability to solve constrained nonlinear least squares problems where the objective function is of the form

$$f(x,p) = r(x,p)^T r(x,p) \quad (44)$$

where $r(x,p)$ is a vector of residuals. The gradient and Hessian of f are

$$\nabla_x f(x,p) = 2 \nabla_x r(x,p)^T r(x,p) \quad (45)$$

$$\nabla_x^2 f(x,p) = V + U \quad (46)$$

$$V = 2 \nabla_x r(x,p)^T \nabla_x r(x,p) \quad (47)$$

$$U = 2 \sum r_i(x,p) \nabla_x^2 r_i(x,p) \quad (48)$$

Unconstrained nonlinear least squares problems can by solved using Gauss' method which calculates a search direction from

$$s = V^{-1} \nabla_x f \quad (49)$$

If the residuals go to zero or $\nabla_x^2 r_i(x,p) = 0$ the method converges quadratically, otherwise it converges linearly. Gauss' method can be improved by approximating the residual Hessian

549

(U) using variable metric updates, and then calculating the search direction from

$$s = (V+U)^{-1} \nabla_x f \qquad (50)$$

A discussion of nonlinear least squares problem can be found in references 3,7,13,20.

NLP2 and NLP3 use a robust least squares method to calculate the search direction as in eq. (50), where a constrained restricted Hessian is approximated. The calculation of the constrained nonlinear least squares search direction NLP2 and NLP3 are described in references 3 and 20. Testing on analytic problems has found that when the least squares options are used, problems are solved in 1/3 fewer function evaluations than standard versions.

The GTL input for a least squares objective function is similar to that for a standard problem. A sample input for a 10-residual problem is

```
OBJFTN  OBJFNM2   (RES(1) UPTO RES(10))
        OBJSKL - 1.E-3
        OBJTOL - 1.E-4
        PGDTOL - 1.E-4
```

Where OBJFNM2 is the objective function model for least squares problems. RES(1) UPTO RES(10) defines the residuals for the problem.

A Comparison of The GRG and GPG Methods

A discussion of some of the similarities for GRG and GPG methods[23,24] concludes the methods are first order root equivalent. This section points out some differences in restricted Hessian, search direction, and constraint solving that affect the performance of GRG and GPG methods.

The convergence of the GRG and GPG methods, are driven by the convergence of the restricted Hessian approximation when variable metric updates are used. In the GRG method, the reduced Hessian is

$$H_r = \begin{bmatrix} J^{-1}C \\ I \end{bmatrix} H \, [\, [J^{-1}C]^T \ I \,] \qquad (51)$$

In the GPG algorithm the projected Hessian is

$$H_p = [Q_2]^T H \, [Q_2] \qquad (52)$$

Where H is the Hessian of the Lagrangian.

Good performance is expected from variable metric updates when the Hessian they are approximating is well conditioned. Additionally for well conditioned Hessians, variable metric updates are less sensitive to exact line searches. Studies at The CorporationAerospace have shown that the condition number of the projected Hessian is generally lower than the condition number of the reduced Hessian. Thus the GPG method has the advantage of approximating a restricted Hessian with a better condition number than the GRG method. The conditioning of the reduced Hessian and search direction for the GRG method are affected by the set of constraint solving variables.

The GRG and GPG methods use different methods for returning to the constraint surface in the line search. Figure 4 illustrates the direction used to return to the constraints for a two variable problem with one nonlinear equality constraint. For the given search direction p, the GRG method searches along S1 when x_1 is the constraint solving variable and along S2 when x_2 is the constraint solving variable. Searching on S1 may be more efficient than searching on S2, depending on the constraints, thus the GRG method can be influenced by the choice of constraint solving variables. The search direction S3 is normal to the constraint surface at the start of the line search and is used by the GPG-type code. S3 represents the steepest descent direction for returning to the constraint surface. When secant updates are applied to the search to return to the constraint surface the search directions are modified.
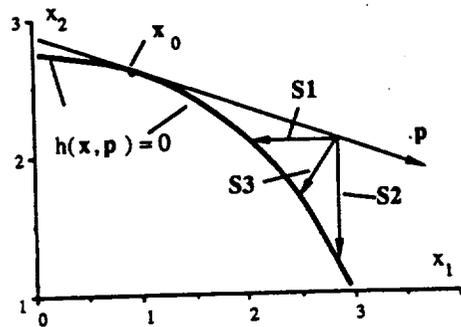
Figure 4: Search Directions for Constraint Solving.

The choice between a GPG and GRG code is based on scaling, constraint nonlinearity, starting point location, and size of the problem. For small dense problems with very nonlinear constraints the GPG method is generally prefered, however on some poorly scaled problems the GRG method may perform better than the GPG method. In our experience the GRG method is sometimes superior to the GPG method when the starting point is far from the optimum. For large sparse problems the GRG method is prefered because it can be programmed to exploit the sparsity, whereas the GPG method generally cannot.

POST OPTIMALITY ANALYSIS

This section describes the post optimality capability in GTS. An optimization is not considered complete until some post optimality analysis is performed. The POSTOP operator contains modules for solution examination, parameter sensitivity analysis, and automatic scaling. Several levels of post optimality analysis are available. A more detailed discussion of POSTOP and its applications can be found in references 2 and 18.

Solution Examination

The POSTOP operator is used to verify that the problem has been solved, and checks if the second order Kuhn-Tucker conditions are satisfied. Several levels of analysis are available.

The first level of POSTOP uses first order gradient information and prints the scaled and unscaled active constraint Jacobian, their singular values, and an estimate of their condition numbers. First level analysis does not require any extra function evaluations, it uses gradients obtained from the optimization algorithm.

First level analysis calculates the Lagrange multipliers using scaled and unscaled gradients. The Lagrange multiplier estimates calculated with the scaled gradients are transformed into unscaled multiplier estimates equation (20), are closer to the true values of the Lagrange multipliers when problems did not converge with $\nabla_x L = 0$ and proper scale weights have been specified. If the problem has been solved tightly, the Lagrange multipliers calculated using unscaled gradients and using scaled gradients and then transformed by equation (20) match to several decimal places[18].

The second level of POSTOP calculates the projected Hessian, its eigenvalues and condition number to verify that the point returned by the optimization algorithm is a local minimum. The analysis is calculated in both scaled and unscaled units. The projected Hessian is a $(n-m_A)$ by $(n-m_A)$ matrix which is usually small for trajectory optimization problems because there are normally only a few degrees of freedom at the solution. Thus only a few extra function evaluations are required by the second level of post optimality analysis. The user can specify that error control algorithms[19] be used to calculate the projected Hessian. When the scaled projected Hessian is ill-conditioned (e.g., has very small eigen values), the solution can be imprecise.

When the projected Hessian is calculated, the constrained Newton correction is calculated[10]. When the Kuhn-Tucker conditions are satisfied, the correction is small. The objective

function and constraints are evaluated using this correction. This locates a slightly more accurate solution than that obtained by the optimization operator for problems that have loosely converged.

Parameter Sensitivity Analysis

Parameter sensitivity analysis is an important area of optimization that has often been overlooked by designers. Parameter sensitivity analysis can assess the effect of variation in input parameters on the optimal solution[8]. An example is to calculate df^*/dp_i the partial of the optimum objective function (note the * is used to indicate optimality) with respect to variations in input parameters.

$$\frac{df^*}{dp_i} = \frac{\partial f}{\partial p_i} + \sum_{j=1}^{m} \lambda_j \frac{\partial c_j}{\partial p_i} \qquad (53)$$

This requires the partials of the objective function and constraints with respect to p_i. This means extra trajectories must be simulated. It also requires the Lagrange multipliers, which can be obtained using first level post optimality analysis.

Figure 4 illustrates the payload capability versus altitude of the orbit for a given mission. This curve can be generated using parameter sensitivity analysis. Mission planners can use the information on payload capability for a vehicle to determine the range of mission orbits attainable for a given payload weight.
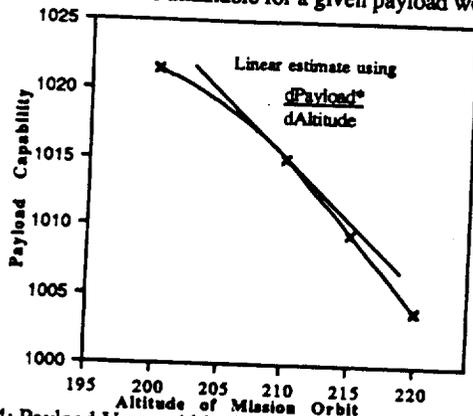
Figure 4: Payload Versus Altitude for generic Two Stage Booster

Parameter sensitivity derivatives can be used to efficiently conduct parametric performance studies, or assess the effect of changes in the weight manifest on vehicle performance.

If the partial of the solution x^* with respect to a parameter is required, then the Hessian of the Lagrangian must be approximated. The third level of POSTOP calculates the Hessian of the Lagrangian.

Automatic scaling

Poor scaling is a major cause of poor performance from NLP algorithms. In trajectory optimization there are no fool proof rules for choosing a good set of scale factors a priori; however, once a good set of scale factors is identified for a particular optimization problem, the scale factors improve the performance on similar problems (e.g., new performance estimate is required for an updated weight manifest).

Automatic scaling routines have been proposed[1]. These algorithms were based on obtaining a set of scale factors that yields a well conditioned constraint Jacobian. Hallman[17] developed an algorithm to automatically scale the constraint Jacobian for trajectory optimization problems, this routine yielded mixed results. On some problems it improved the performance and on others it impeded the convergence of the codes. This occurred when the constraint Jacobian was scaled without regard to the restricted Hessian. It can cause the restricted Hessian to become poorly scaled and ill-conditioned.

This presents a situation where the optimal scaling cannot be determined until the restricted Hessian is known and the restricted

Hessian is not know until the solution is found. At this point the optimal scaling does not help solve the problem. The optimal scaling can be used when similar problems are solved.

POSTOP contains algorithms which attempt to scale the problem so the scaled projected Hessian resembles the identity matrix, and the active constraint Jacobian is well conditioned[16]. This scaling can yield a dramatic improvement in the performance of a GPG code when used to solve similar problems. The scaling obtained from POSTOP may or may not improve the performance of a GRG code, depending on the basis that is used and the condition number of the reduced Hessian.

SUMMARY

The effort put into development of robust optimization software in the GTS system has led to efficient codes that require a minimum number of simulated trajectories be flown. The poorly scaled nature of trajectory optimization problems has required that scaling and accurate linear algebra be used for good algorithm performance.

The development of optimization algorithms has reduced the cost of estimating performance of boosters, upper stages, and reentry vehicles, and allowed many trade studied to be conducted.

The POSTOP capability (solution verification / parameter sensitivity analysis / and automatic scaling algorithms) are new features that have begun to reduce the time required to solve problems. The automatic scaling can drastically reduce the effort required to solve problems.

Optimization is an integral part of the analysis software in GTS. The flexibility of GTS has led to the solution of complicated trajectory optimization problems, and has allowed new combinations of variables to be studied to compare alternate schemes for trajectory / vehicle design.

REFERENCES

1. Balachandran, S., "Survey of Techniques for Scaling Nonlinear Programs," Proceedings of ASME Design Conference, 1985.

2. Beltracchi, T. J., "A Decomposition Approach to Solving the Allup Trajectory Optimization Problem," AIAA Paper 90-0469, Jan. 1990.

3. Betts, J. T., and Hallman, W. P., 1989, "NLP2 Optimization Algorithm Documentation," The Aerospace Corporation, TOR -0089(4464-06)-1, 28 August 1989.

4. Coleman, T. F., and Sorensen, D. C., 1984, "A Note on the Computation of an Orthonormal Basis for the Null Space of a Martix," in Mathematical Programming 29 pp 234-242.

5. DeBilzan C. C., "An Overview: The GTS Software System as Applied to Space Shuttle and Interim Upper Stage Studies," Presented at the Simulation and the Space Shuttle Conference, May 4-6, 1977, Downey, California.

6. DeBilzen, C. C., "SIZE: The Aerospace Corporation's Modular Vehicle Design Program," AIAA Paper 75-1275, 1975.

7. Dennis and Schnable, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice Hall 1983 .

8. Fiacco, A. V., Introduction to Sensitivity and Stability Analysis in Nonlinear Programming, Academic Press 1983.

9. GTS, "The Generalized Trajectory Simulation System," Vol 1 - 5, Report No TR-0076(6666)-1, The Aerospace Corporation, El Segundo, CA, 1976

10. Gabay, D., "Reduced Quasi-Newton Methods with Feasibility Improvement for Nonlinearly Constrained Optimization, Mathematical Programming Study 16, Ed. A. G. Buckey and J. L. Goffin, North Holland Publishing CO. p18-44, 1982.

11. Gabriele, G. A., and Ragsdell, K. M., "OPT - A Nonlinear Programming Code in FORTRAN IV," Purdue Research Foundation, W. Lafayette IN. 1976.

12. Gabriele, G. A., and Beltracchi, T. J., "Users Manual OPT3.2, A FORTRAN 77 Implementation of the Generalized Reduced Gradient Algorithm, 1987, Design Automation Laboratory, Rensselaer Polytechnic Institute, Troy NY.

13. Gill, P. E., Murray, W., and Wright, M. H., Practical Optimization, Academic Press 1981.

14. Golub, G. H., and VanLoan, C. F., Matrix Computations, Johns Hopkins 1985.

15. Hallman, W. P., "A Software Package for Post Optimality Analysis of the Nonlinear Programming Problem," The Aerospace Corporation ATM No. 86(6464-06)-5, 29 April 1986. Not available for external distribution.

16. Hallman, W. P., "Optimal Scaling Techniques for the Nonlinear Programming Problem," The Aerospace Corporation ATM No 86(4464-06)-7, June 1989, Not available for external distribution.

17. Hallman, W. P., "Optimization Update For GTS System 35E," The Aerospace Corporation IOC A- 88-5752.4-04, August 30 1988, Not available for external distribution.

18. Hallman, W. P., "Sensitivity Analysis for Trajectory Optimization Problems," AIAA Paper 90-0471, Jan 1990.

19. Hallman, W. P., "Numerical Derivative Techniques for Trajectory Optimization," to appear in Third Air Force/NASA Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, San Francisco, CA 1990.

20. Hallman, W. P., "NLP3 Optimization Algorithm Documentation," to appear as Technical Operating Report, The Aerospace Corporation.

21. Luke, R. A., "Computational Efficiency Considerations for High Fidelity Launch Vehicle Trajectory Optimization," in Proceedings of AIAA Guidance, Navigation, and Control Conference, Boston MA, August 1989.

22. Meder, D. S., and McLaughlin, J. R., "A Generalized Trajectory Simulation System," Presented at the 1976 Summer Computer Simulation Conference, July 12-14 1976, Washington D. C.

23. Reklaitis, G. V., Ravindrain A., Ragsdell, K. M., Engineering Optimization Methods and Applications, John Wiley and Sons 1983.

24. Sargent, R. W. H., "Reduced Gradient and Projection Methods for Nonlinear Programming," in Numerical Methods for Constrained Optimization (P. E. Gill and W. Murray Eds.) Academic Press, London England, 1974.

OPTIMUM DESIGN OF A COMPOSITE STRUCTURE WITH PLY-INTERLEAVING CONSTRAINTS

Bo Ping Wang and Daniel P. Costin
Department of Mechanical Engineering
The University of Texas at Arlington
Box 19023
Arlington, Texas 76019
(817) 273-2021

INTRODUCTION

The application of composite materials to aircraft construction has provided the designer with increased flexibility. The orientation of plies can be tailored to provide additional aeroelastic performance unobtainable with an isotropic material. A tailored laminate is made up of plies of several orientations, usually 0°, 45°, -45°, and 90°. The direction of the 0° plies, does not need to be oriented with the leading edge, but can be varied to obtain a wide variety of structural properties. Also, the number of plies of each orientation varies from one zone to another on the planform. Thus, a thick laminate with mainly 0° plies may form the root zone, and a thinner laminate with mainly ±45° plies may form the the leading edge zone. Examples of tailored aircraft structures can be found in the publications of Rogers, Braymen, and Shirk [1], Schmit and Fleury [2] and Johnson and Neill [3]. Tailored laminates were designed using complicated optimization programs. Unfortunately, many tailored designs must be modified before they are manufactured. The modification adds weight and decreases performance. One type of modification is ply interleaving, an overlap of plies between zones on the laminate. These interleaves are added to ensure that zones with varying ply percentages can be connected without loss of strength.

In this paper, the constraints needed to eliminate interleaves in the laminate optimization process will be described and implemented in a structural optimization problem. The method used has the potential to prevent changes to composite laminates late in the design cycle.

Examples of tailored laminates that require interleaving are not difficult to find. Schmit and Fleury [2] published a delta wing design shown in Figure 1. The planform of the wing is shown as it would be seen from above. The numbers listed for each zone are thicknesses of 0°, ±45°, and 90° plies. An interleave for a zone is required whenever an adjacent zone has more plies of one orientation and fewer of another. Thus Section A-A of Figure 1 (not drawn to scale) would have a 90° ply in Zone 1 that would have to be interleaved into Zone 2. Other pairs of zones require interleaving, such as 3-6, 6-9, 9-11, 7-10, 10-12, and 12-14. The extra weight and the change in stiffness caused by interleaving may adversely affect this design.

Another example of a tailored laminate is the Intermediate Complexity Wing described in the ASTROS (Automated Structural Optimization) Applications Manual [3]. The top view of this wing is shown in Figure 2. The numbers indicated are the numbers of 0°, 90°, +45, -45 plies in the laminate at each zone. Figure 2 also shows Section A-A of the laminate. Again, an interleave must be added to prevent the thickness from dropping to five plies from six. In summary, neither of the algorithms used in Reference [2] and [3] attempted to eliminate the possibility of interleaves, and laminates with interleaves are a common consequence.

SECTION A - A

② ①

0	
0	
0	0
±45	±45
90	90
±45	90
±45	±45
0	0
0	
0	

(0°,±45°,90°)

⑯
(6, 2, 1)

⑮
(17, 9, 1)

⑬
(2, 6, 1)

⑭
(41, 12, 1)

⑪
(8, 9, 1)

⑫
(75, 11, 1)

⑧
(1, 1, 1)

⑨
(16, 8, 1)

⑩
(116, 8, 1)

⑤
(3, 2, 1)

⑥
(23, 6, 1)

⑦
(151, 7, 3)

A

①
(2, 2, 2)

②
(6, 3, 1)

③
(30, 5, 1)

④
(187, 7, 3)

A
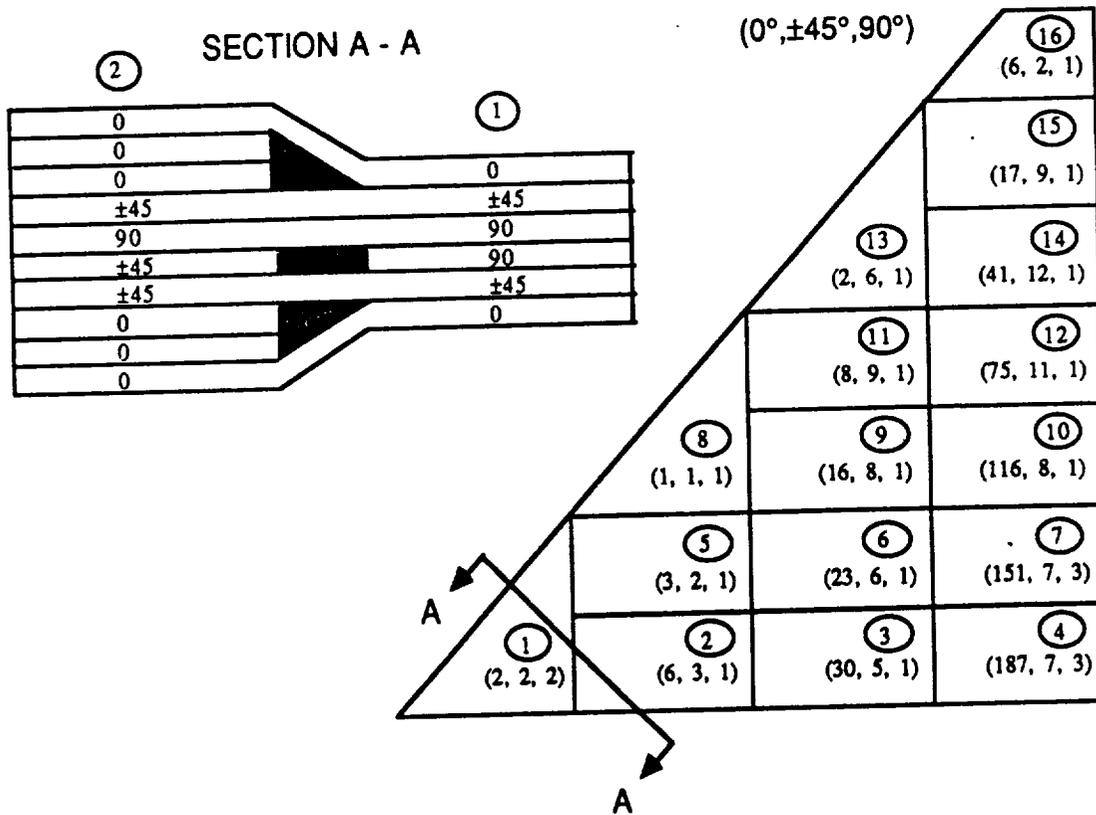
Figure 1. The delta wing designed by Schmit and Fleury required interleaves at several locations,
including Section A-A.

(0°, 90°, +45°, -45°)

32 (3, 1, 1, 1)
31 (4, 1, 2, 1)
24 (12, 1, 3, 1)
30 (4, 1, 2, 1)
23 (10, 1, 2, 1)
29 (3, 1, 1, 1)
16 (18, 1, 4, 1)
22 (9, 1, 1, 1)
28 (3, 1, 1, 1)
15 (17, 1, 4, 1)
21 (7, 1, 1, 1)
27 (2, 1, 1, 1)
8 (21, 1, 4, 1)
14 (16, 1, 2, 1)
20 (6, 1, 1, 1)
26 (2, 1, 1, 1)
7 (20, 1, 1, 1)
13 (13, 1, 1, 1)
19 (4, 1, 1, 1)
25 (2, 2, 2, 2)
6 (16, 1, 1, 1)
12 (10, 1, 1, 1)
18 (2, 1, 1, 1)
17 (1, 1, 1, 2)
5 (12, 1, 1, 1)
11 (6, 1, 1, 1)
9 (1, 1, 1, 1)
4 (9, 1, 1, 2)
10 (4, 1, 1, 1)
1 (2, 1, 1, 2)
3 (6, 1, 1, 2)
2 (3, 1, 1, 1)

SECTION A - A

①	②
0	0
45	45
-45	90
90	0
-45	-45
0	0

A   A
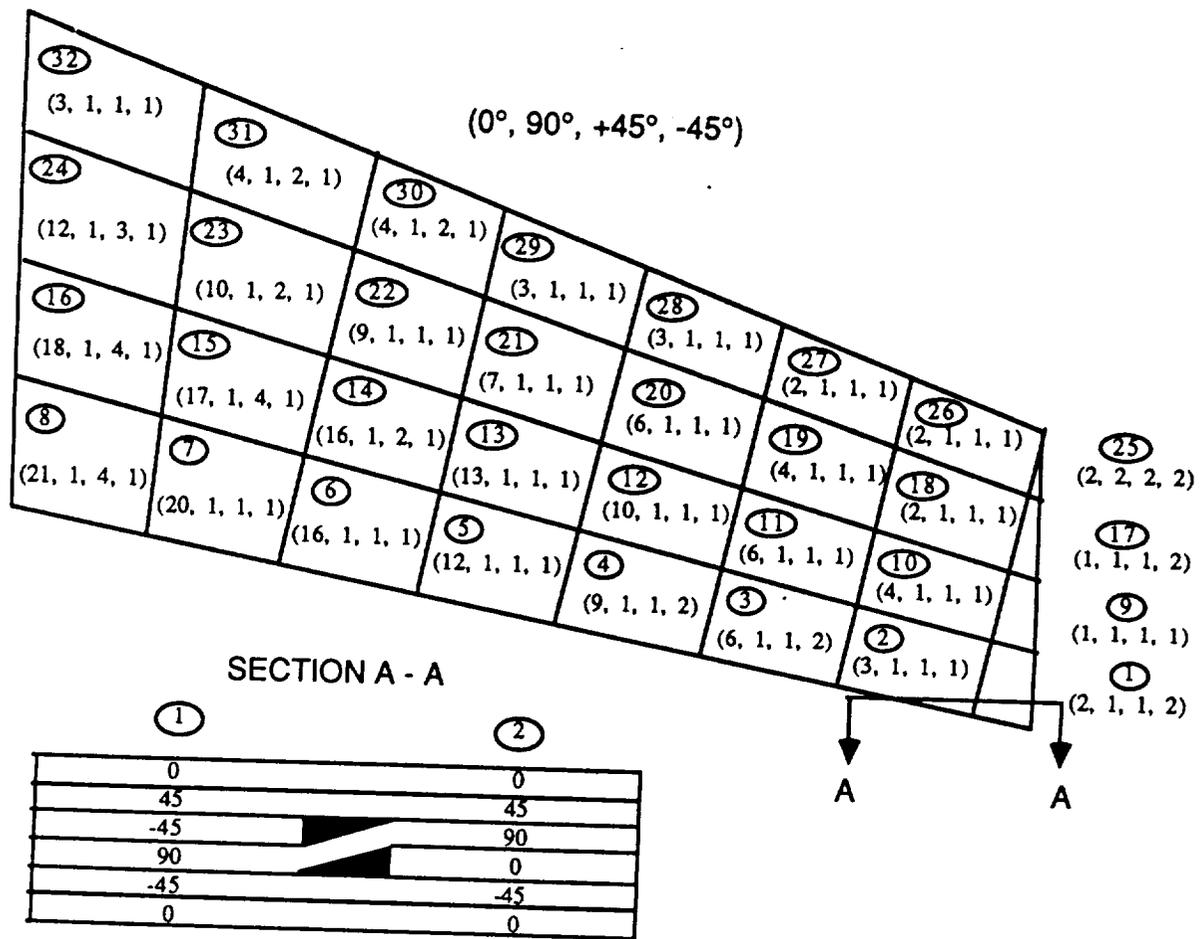
Figure 2. The Intermediate Complexity Wing was designed using ASTROS, but interleaves were required in several locations, including Section A-A.

A way to avoid the interleaving problem is to impose constraints on the thicknesses in the design problem so that all plies in a thin zone are continuous into a zone that is thicker. This method is simple in concept because the constraint is linear function of the design variables. It requires, however, that the design variables be a constant thickness for a zone on the laminate. It precludes the use of an optimization scheme in which the thicknesses are described by a continuous shape function. This method requires multiple optimization runs, and often requires a larger number of design variables than a continuous shape function optimization scheme.

Recently, the Air Force has sponsored the development of ASTROS. This multidisciplinary program can tailor composite structures for a variety of constraints. It uses the finite element method for structural response, providing potentially greater accuracy than the simplified plate analysis used in Reference [1]. ASTROS was used in this study because of its modularity. It allows new subroutines to be added without modifying the existing ASTROS code. A subroutine was written to apply direct constraints on the design variables, and it was used to apply interleaving constraints.

MATHEMATICAL DESCRIPTION OF INTERLEAVING CONSTRAINTS

To eliminate interleaves, the individual orientation thicknesses must increase if the total thickness increases at a zone boundary. In the following equations, t_1 and t_2 are the total thicknesses of two adjacent zones, and $(t_n)_{\theta°}$ are the thicknesses of ply orientation θ.

if $t_1 \leq t_2$ then

$$(t_1)_{0°} \leq (t_2)_{0°}$$
$$(t_1)_{45°} \leq (t_2)_{45°}$$
$$(t_1)_{-45°} \leq (t_2)_{-45°}$$
$$(t_1)_{90°} \leq (t_2)_{90°}$$

$$(1)$$

if $t_1 \geq t_2$ then

$$(t_1)_{0°} \geq (t_2)_{0°}$$
$$(t_1)_{45°} \geq (t_2)_{45°}$$
$$(t_1)_{-45°} \geq (t_2)_{-45°}$$
$$(t_1)_{90°} \geq (t_2)_{90°}$$

The difficulty with this type of constraint is that it is conditional. Each condition, if feasible, will give a unique optimum weight. The condition that will give the lowest optimum is not known before the optimization runs begin. With a multiple zone composite laminate, the number of combinations of constraint conditions can be large. Consider a four zone laminate with two zones along the span and two zones along the chord. The sixteen possible constraint combinations are as follows.

1:	$t_1 \geq t_2$	$t_1 \geq t_3$	$t_3 \geq t_4$	$t_2 \geq t_4$
2:	$t_1 \geq t_2$	$t_1 \geq t_3$	$t_3 \geq t_4$	$t_2 \leq t_4$
3:	$t_1 \geq t_2$	$t_1 \geq t_3$	$t_3 \leq t_4$	$t_2 \geq t_4$
4:	$t_1 \geq t_2$	$t_1 \geq t_3$	$t_3 \leq t_4$	$t_2 \leq t_4$
5:	$t_1 \geq t_2$	$t_1 \leq t_3$	$t_3 \geq t_4$	$t_2 \geq t_4$
6:	$t_1 \geq t_2$	$t_1 \leq t_3$	$t_3 \geq t_4$	$t_2 \leq t_4$
7:	$t_1 \geq t_2$	$t_1 \leq t_3$	$t_3 \leq t_4$	$t_2 \geq t_4$
8:	$t_1 \geq t_2$	$t_1 \leq t_3$	$t_3 \leq t_4$	$t_2 \leq t_4$
9:	$t_1 \leq t_2$	$t_1 \geq t_3$	$t_3 \geq t_4$	$t_2 \geq t_4$
10:	$t_1 \leq t_2$	$t_1 \geq t_3$	$t_3 \geq t_4$	$t_2 \leq t_4$
11:	$t_1 \leq t_2$	$t_1 \geq t_3$	$t_3 \leq t_4$	$t_2 \geq t_4$
12:	$t_1 \leq t_2$	$t_1 \geq t_3$	$t_3 \leq t_4$	$t_2 \leq t_4$
13:	$t_1 \leq t_2$	$t_1 \leq t_3$	$t_3 \geq t_4$	$t_2 \geq t_4$
14:	$t_1 \leq t_2$	$t_1 \leq t_3$	$t_3 \geq t_4$	$t_2 \leq t_4$
15:	$t_1 \leq t_2$	$t_1 \leq t_3$	$t_3 \leq t_4$	$t_2 \geq t_4$
16:	$t_1 \leq t_2$	$t_1 \leq t_3$	$t_3 \leq t_4$	$t_2 \leq t_4$

$$(2)$$

For a twenty zone laminate with five zones along the span and four zones along the chord, over 2 billion constraint combinations are required. One way to solve this problem is to find the optimum solution for all constraint combinations. The best of these solutions would be the true constrained optimum. It can be proven, however, that a limited subset of the sixteen combinations can be solved, while still providing the true constrained optimum. This method is called the branch and bound method. It was originally developed by Balinski [4] in 1965 to solve linear programming problems with integer design variables. Balinski's method starts with a continuous variable optimization, and constraints similar to Equation 2 are added to close in on an integer solution. Branch and Bound was subsequently modified and used for Civil engineering optimization problems, which often require the use of standard cross-sections for beams or an integer number of reinforcing rods for concrete [5].

For the laminate optimization problem the top of the Branch and Bound tree is the unconstrained optimum, as shown in Figure 3. The term unconstrained refers to the constraints of expression (1). The tree branches to the constraint $t_1 \leq t_2$, and to the constraint $t_1 \geq t_2$. Additional branches are added so that the bottom of the branches correspond to the constraint combinations of Equation (2). A single constraint combination is solved, and a temporary optimum solution with weight f^* is found. Now the other branches of the tree are explored, starting with nodes near the top. If any of these nodes near the top have

556

an optimum weight $f_n > f^*$ then all of the nodes that are positioned beneath it have an optimum weight $f_n > f^*$. This method eliminates most of the tree if a good guess at the best constraint combination can be found.
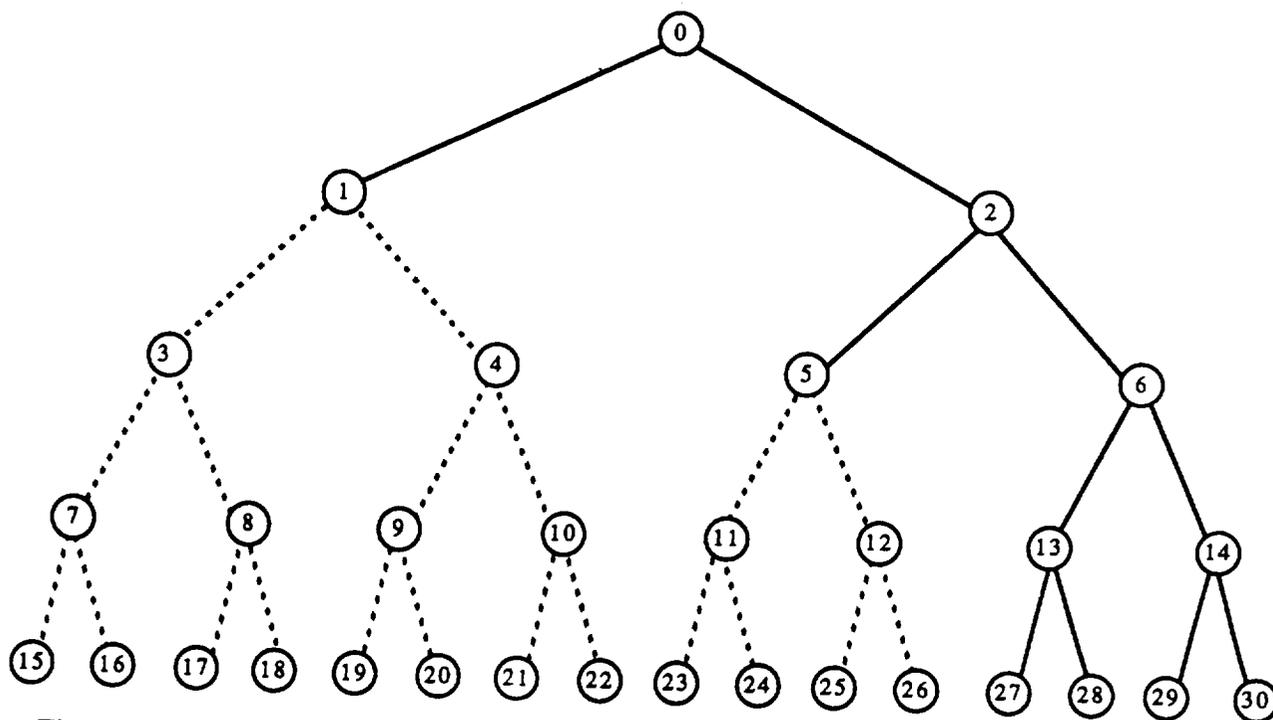
Figure 3. The branch and bound method reduces the number of constraint combinations that must be analyzed.

ASTROS MODEL OF THE SIMPLE WING STRUCTURE

The ASTROS Simple Wing Structure (SWS) was chosen to evaluate the branch and bound method. The SWS, shown in Figure 4, has four zones of unique thickness on the wing skin. Each zone has four design variables, corresponding to the thicknesses of $0°$, $45°$, $-45°$, and $90°$ plies. The top and bottom of the wing are symmetric. A simulated aerodynamic load was applied to the model, and the resulting optimum design was constrained to have a tip displacement less than 10 inches, and a wash-out tip twist of greater than 5% of the chord. Tsai-Wu stress constraints were also applied to the composite skins. The thickness of the aluminum understructure was not allowed to vary.

TOP VIEW

20

60

AERODYNAMIC
LOAD

SIDE VIEW

1

DIMENSIONS IN INCHES
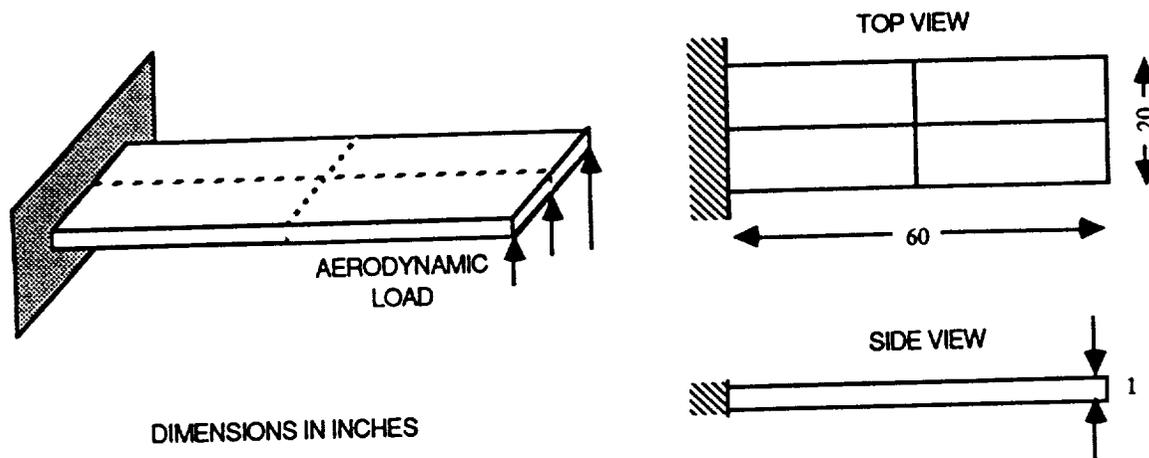
Figure 4. The Simple Wing Structure (SWS) was used to test the branch and bound method because it has only eight membrane elements modeling the skins, and requires only sixteen design variables.

RESULTS

Each circle shown on the branch and bound tree is called a node. The first optimization run is Node 0. This run has aeroelastic and stress constraints, but no constraints on the thicknesses. Node 27 has four thickness constraints, and is the node most likely to be optimum, because the total thicknesses of Node 0 satisfy the Node 27 constraint combination. For this reason, Node 27 is run to obtain a starting constrained optimum weight, as shown in Figure 5. The weights of the two designs are show below.

$$f_0 = 9.36 \text{ lbs}$$
$$f_{27} = 9.77 \text{ lbs}$$

This indicates that a 4.4% weight penalty is associated with constraint combination of Node 27. The resulting ply thicknesses, shown in Figure 6, reveal the design changes required to satisfy the constraint. The $-45°$ plies in Zone 2 and Zone 3 are extended into Zone 1, and some $0°$ plies are removed from Zone 1 and Zone 4 to minimize weight and satisfy the aeroelastic constraints.
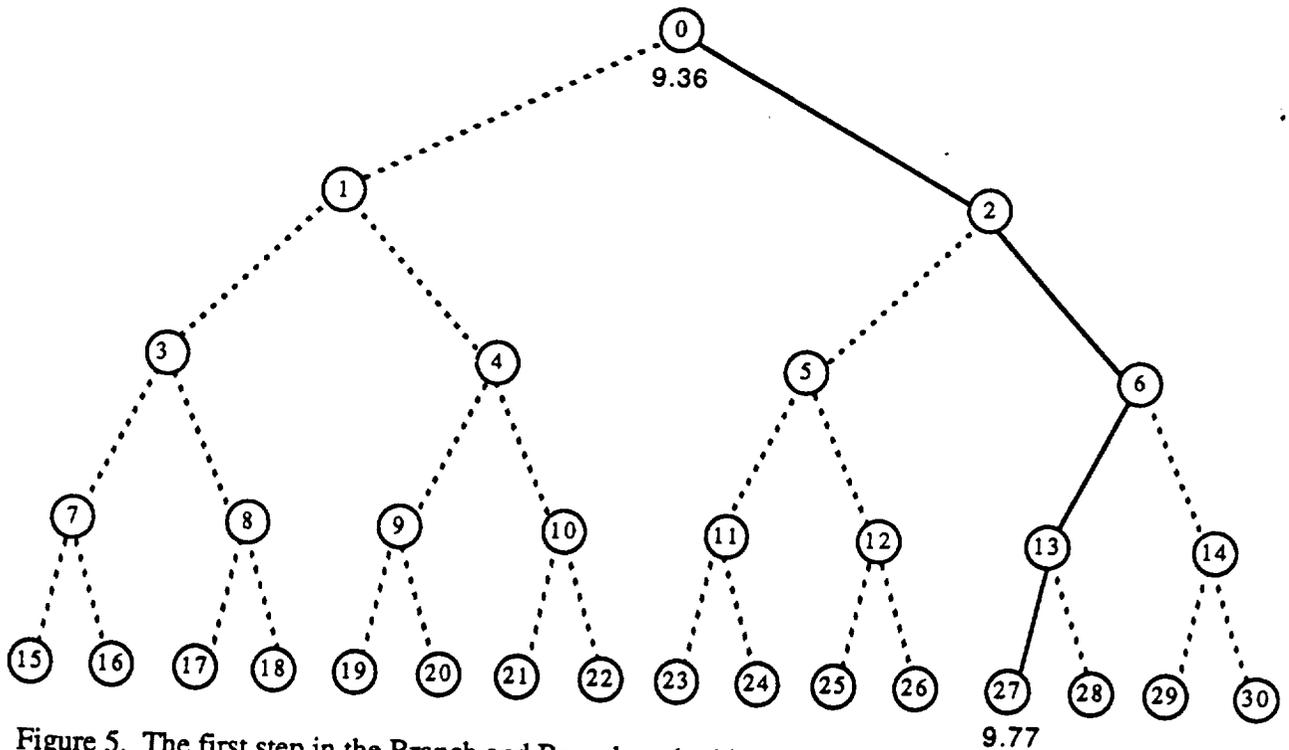
Figure 5. The first step in the Branch and Bound method is to calculate the most likely feasible solution.
(f*=9.77)
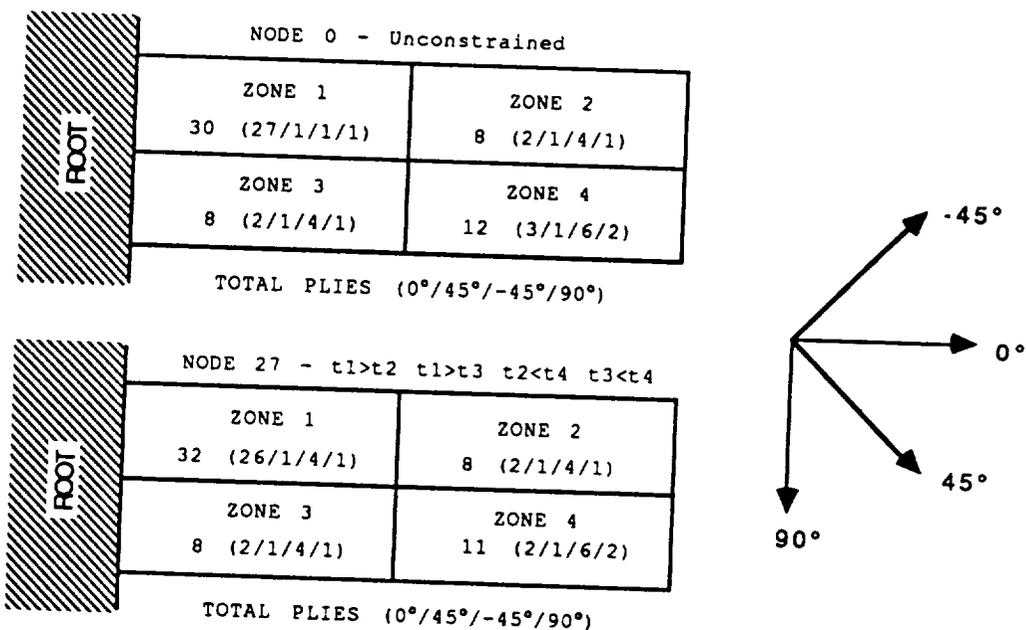
Figure 6. Ply thicknesses of Node 0 and Node 27 show how the thicknesses change to satisfy interleaving constraints.

The next step in the analysis is to check all the other branches of the tree to find out if a better constrained optimum exists. Figure 7 shows the results of the optimization runs corresponding to the other branches. Node 1 converged to an optimum weight of 19.15 lbs. This weight was greater than f*, and additional constraints on this problem would only increase weight. Thus all branches of the tree extending from Node 1 were disregarded. Next, the optimization run of Node 5 was performed, and it

559

converged to an optimum weight of 19.25 lbs. Since the optimum weight of Node 5 was greater than f*, all branches extending from Node 5 were disregarded. Next, the optimization run of Node 14 was performed, and it converged to an optimum value of 10.10 lbs. Since the optimum weight of Node 14 was greater than f*, all branches extending from Node 14 were disregarded. Next, the optimization run of Node 28 was performed, and it converged to an optimum weight of 10.09 lbs. When Node 28 was completed the entire tree had been spanned, and lowest constrained optimum was obtained with Node 27. Node 27 is the true optimum solution.
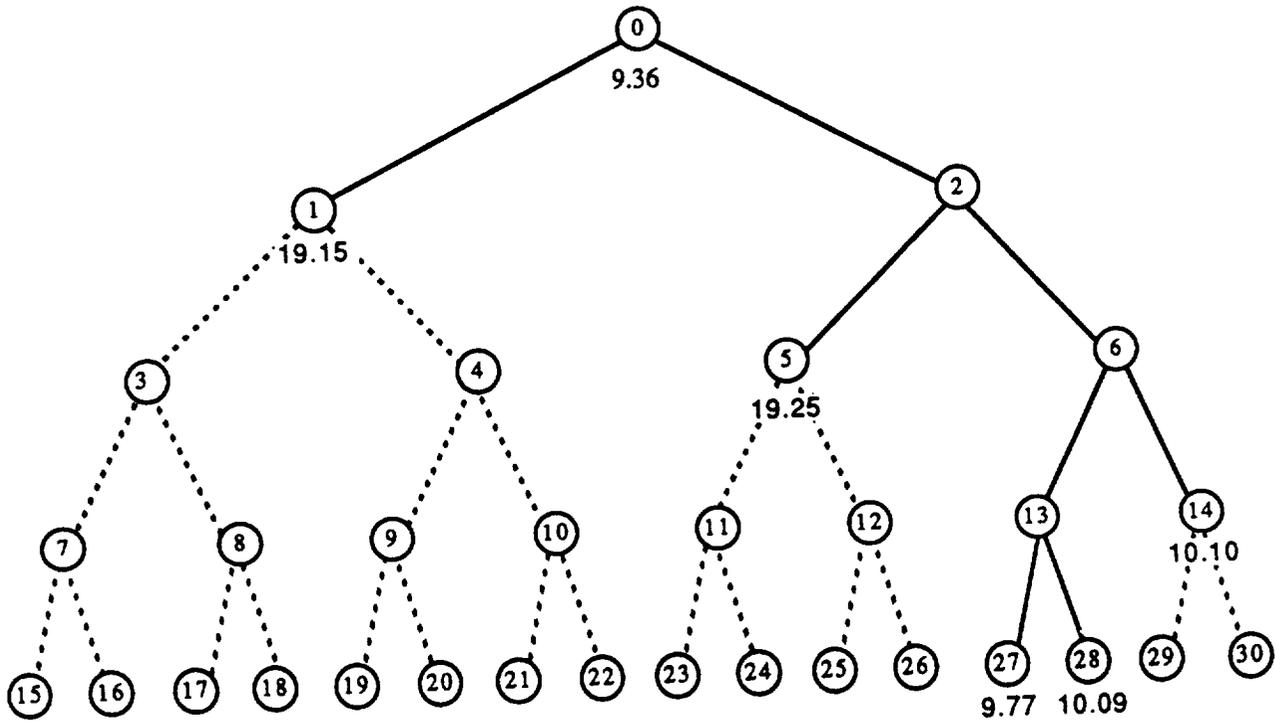
Figure 7. Checking all the branches required only six optimization runs, because most nodes cannot satisfy all the design constraints without a significant increase in weight.

CONCLUSIONS

The branch and bound method of optimization is time-consuming, but it rigorously finds the true optimum solution. The major benefit is that it reveals a more practical method of reaching a feasible solution that may be very close to the optimum. The practical method uses a combination of thickness constraints that corresponds to the total thicknesses of the unconstrained solution. This simplified way of obtaining a manufacturable design may be appropriate for large optimization problems.

The resulting design using ASTROS with interleaving constraints is slightly different from the design that would be obtained by a heuristic method. For both methods, interleaved plies are extended into thicker zones to satisfy interleaving constraints. ASTROS then takes other plies out of the laminate that are not needed because of the presence of the extended plies.

Optimization runs converged in five to eleven iterations. Convergence was more difficult for runs that started in an infeasible domain.

Modifications to ASTROS were straight-forward due to the flexibility of the code and the comprehensive documentation.

ACKNOWLEDGEMENTS

REFERENCES

[1] W. A. Rogers, W. W. Braymen, and M. H. Shirk, "Design, Analyses, and Model Tests of an Aeroelastically Tailored Lifting Surface," J. Aircraft, Vol. 20, No. 3, March 1983, pp. 208-215.

[2] L. A. Schmit and C. Fleury, "Structural Synthesis by Combining Approximation Concepts and Dual Methods," AIAA Journal, Vol. 18, No. 10, October 1980, pp. 1252-1260.

[3] E. H. Johnson and B. J. Neill, "Automated Structural Optimization System 'ASTROS' Vol. 3- Applications Manual," Dec. 1988, Air Force Wright Aeronautical Laboratories, Wright Patterson AFB, Ohio.

[4] M. L. Balinski, "Integer Programming: Methods, Uses, Computations," Management Science, Vol. 12, No. 3, November 1965, pp. 253-313.

[5] K. F. Reinschmidt, "Discrete Structural Optimization," Journal of Structural Division, ASCE, Vol. 97, No ST1, January, 1971, pp. 133-156.

OPTIMAL DESIGN OF CONTROL SYSTEMS FOR
PREVENTION OF AIRCRAFT FLIGHT DEPARTURE

C. Edward Lan and Fuying Ge

2004 Learned Hall
Department of Aerospace Engineering
The University of Kansas
Lawrence, Kansas 66045

ABSTRACT

Control system design for general nonlinear flight dynamic models is considered through numerical simulation. The design is accomplished through a numerical optimizer coupled with analysis of flight dynamic equations. In the analysis, the general nonlinear flight dynamic equations with nonlinear aerodynamics are numerically integrated; and the dynamic characteristics needed in the optimization process are then identified from the dynamic response. To assure a reasonable solution, the initial input values of the design variables are estimated through a sensitivity analysis. To demonstrate the method, the pitch departure prevention for an F-16 configuration is demonstrated.

INTRODUCTION

At high angles of attack, the aerodynamic forces and moments are, in general, time-dependent and nonlinear functions of motion variables. In addition, the aerodynamic, kinematic, and inertial coupling phenomena are important to the high angle-of-attack flight dynamics of modern aircraft. One feature of a high-alpha control system is the simultaneous utilization of several control surfaces or devices. Based on these considerations, a number of high angle-of-attack control concepts have emerged (refs. 1-4). However, to perform the detailed control system design, a suitable method must be capable of incorporating these coupling phenomena with considerations of time-dependent, nonlinear aerodynamic forces and moments. In addition, the method must be capable of handling multiple input and output. A current approach to solve this problem is by extensive piloted simulation (ref. 5).

Methods in optimal control theory represent possible approaches to solving these problems under consideration. However, computational methods in existence require local linearization of dynamic equations and aerodynamics. Another alternative is to apply numerical optimization techniques without linearization as they are frequently used in structural and aerodynamic designs of large systems. A similar approach has also been used in other control applications in ref. 6.

In the present method, a numerical optimization technique based on conjugate gradients and feasible directions (ref. 7) is coupled with an analysis method which is to obtain the numerical solutions of the nonlinear six-degree-of-freedom dynamic equations. This analysis method is to provide information needed in the design process. Since the analysis method can deal with nonlinearities in the dynamics and the aerodynamics and with any general constraints on the control system configuration, the control system designed with a numerical optimization technique can be very realistic and effective.

To demonstrate the present method, design of control systems to prevent flight departure at high angles of attack in a maneuver will be considered. Since results of optimizing a nonlinear system tend to depend on the initial data to start the design process, how appropriate initial values of design variables can be chosen will be illustrated in this paper. Some results of the present method, without considering the effect of initial values of design variables, have been presented in ref. 8.

DESIGN METHODOLOGY

At high angles of attack, the main flight dynamic problems are pitch departure and lateral-directional instabilities. These problems are aggravated by the reduced control effectiveness. From a mathematical point of view, the problem involves nonlinear 6-DOF flight dynamic equations with nonlinear aerodynamic forces and moments. The present method is based on numerical integration of the following nonlinear dynamic equations:

$$m(\dot{u} - vr + wq) = mg_x + F_{A_x} + F_{T_x} \tag{1a}$$

$$m(\dot{v} + ur - wp) = mg_y + F_{A_y} + F_{T_y} \tag{1b}$$

$$m(\dot{w} - uq + vp) = mg_z + F_{A_z} + F_{T_z} \tag{1c}$$

$$I_{xx}\dot{p} - I_{xz}\dot{r} - I_{xz}pq + (I_{zz} - I_{yy})rq = L_A + L_T \tag{1d}$$

$$I_{yy}\dot{q} + (I_{xx} - I_{zz})pr + I_{xz}(p^2 - r^2) = M_A + M_T \tag{1e}$$

$$I_{zz}\dot{r} - I_{xz}\dot{p} + (I_{yy} - I_{xx})pq + I_{xz}qr = N_A + N_T \tag{1f}$$

$$\dot{\phi} = p + q \sin\phi \tan\theta + r \cos\phi \tan\theta \tag{1g}$$

$$\dot{\theta} = q \cos\phi - r \sin\phi \tag{1h}$$

$$\dot{\psi} = (q \sin\phi + r \cos\phi)\sec\theta \tag{1i}$$

$$\alpha = \tan^{-1}(w/u) \tag{1j}$$

$$\beta = \sin^{-1}(v/\sqrt{u^2 + v^2 + w^2}) \tag{1k}$$

where (u, v, w) are the three linear velocity components of the aircraft; (p, q, r) are the angular velocity components; and (ϕ, θ, ψ) are the Euler angles in roll, pitch, and yaw, respectively. g is the gravitational acceleration, and F's are the external forces, while (L, M, N) are the moments about the (x-, y-, z-) axes. In addition, m is the mass and I_{xx}, I_{xz}, etc., are the moments of inertia. The subscripts A, T denote the aerodynamic and thrust forces and moments, respectively. The aerodynamic forces and moments (F_A, L_A, M_A, N_A), including the control effects, are represented in dimensionless coefficients in a tabulated form as functions of motion variables in this study. The motion variables are (u, v, w, p, q, r). The results of the integrated motion variables are then used by the optimizer (ref. 7) to determine the best strategy of control.

To be more specific, it is desirable to minimize the sideslip (β) in a maneuver. At the same time, to have good tracking ability, the maximum transient α response (α_{max}) and maximum change in yaw angle (ψ_{max}) should be minimized. To prevent spin entry, the maximum yaw rate (r_{max}) should also be as small as possible. Therefore, a possible objective in a roll maneuver is to minimize the following objective function:

$$OBJ = -C_1 P_{max} - C_2 \alpha_{trim} - \frac{C_3}{|\alpha_{max}| + \epsilon} - \frac{C_4}{|\psi_{max}| + \epsilon} - \frac{C_5}{|\beta_{max}| + \epsilon}$$

$$- \frac{C_6}{|r_{max}| + \epsilon} - \frac{C_7}{|\theta_{max}| + \epsilon} - \frac{C_8}{|\phi_{trim}| + \epsilon} \tag{2}$$

subject to various constraints depending on applications. α_{trim} may be used to define the limiting angle of attack to be discussed later for application to an F-16. The constants, $C_1 - C_8$, are weighting factors to be chosen so that all terms have the same order of magnitude. ϵ is taken to be 10^{-7}. The control system

structure must be assumed, with the possible design variables to be determined by the optimizer, being

aileron-rudder interconnect gain (K_{ARI}),

side acceleration feedback gain (K_{ay}),

yaw damper gain (K_r), roll damper gain (K_p),

pitch damper gain (K_q), angle of attack feedback gain (K_α),

normal acceleration feedback gain (K_{az}),

roll control deflection feedback gain (K_δ), etc.

where K_δ is defined in

$$\delta_a = \delta_{a_{max}} - K_\delta p$$

$$(3)$$

Various constraints can be imposed in the calculation. In general, any constraints which affect the results and can be identified from dynamic analysis can be imposed.

NUMERICAL RESULTS

The aerodynamic data for an F-16 configuration are obtained from ref. 9. Since the F-16 is unstable in pitch, design of a pitch control system is of major concern. The control system includes an angle-of-attack/normal-acceleration limiting system. In the α-limiting system, the pitch control deflection (δ_e) due to the α-feedback is defined as

$$\delta_e(\alpha\text{-feedback}) = K_\alpha \alpha - K_c$$

$$(4)$$

The maximum thrust is assumed to be 8100 lbs at 30,000 ft altitude. The assumed control system structure is illustrated in Fig. 1.

A proper choice of the initial values of design variables used in the present method is obtained through a quick sensitivity analysis in the code. That is, the design variables are systematically varied in large increments and their effect on the objective function is then used to determine the proper ranges of values of the design variables for the optimization process. For the F-16 configuration, the results of sensitivity analysis for the design variables K_{ARI}, K_q, and K_α are presented in Fig. 2. The pivotal values around which these parameters are varied are assumed as follows:

$$K_{ARI} = 1.05, \qquad K_{ay} = 16., \qquad K_r = 0.6,$$
$$K_q = 3.6, \qquad K_{az} = 0.3, \qquad K_\alpha = 3.,$$
$$K_c = 5., \qquad K_p = 0.2, \qquad K_\delta = 0.2$$

Based on these results, the starting values of the design variables are then chosen as follows:

$$K_{ARI} = 0.9, \qquad K_{ay} = 18., \qquad K_r = 0.4,$$
$$K_q = 3.8, \qquad K_{az} = 0.5, \qquad K_\alpha = 2.9,$$
$$K_c = 10., \qquad K_p = 0.06, \qquad K_\delta = 0.2$$

Note that the objective function defined in Eq. (2) has a sharp minimum in the space of design variables. Taking $C_1 = 0.008$, $C_2 = 0.05$, $C_3 = 1.0$, $C_4 = 14.0$, $C_5 = 0.01$, $C_6 = 0.008$, $C_7 = 0.7$, and $C_8 = 1.0$, the variation of the objective function is illustrated in Fig. 3.

As an application of the aforementioned procedures, an optimal set of system gains to prevent pitch departure in a roll maneuver for an F-16 configuration will be determined. Note that due to inertial coupling, the presence of roll rate will induce additional pitch-up to make the problem worse. Note that α_{trim} in Eq. (2) is defined as the average angle of attack over the whole time period. It is to be maximized while maintaining stability of the airplane. The roll control is applied between t = 22 and 34 sec. Results shown in Fig. 4 show that no departure has occurred and the final angle of attack is α_{trim} = 24.5 deg. In this case,

$$P_{max} = 62.0 \text{ deg/sec.}$$

The calculation converges after 50 iterations. It is noted that if there is no α-limiting system (K_α = 0, K_c = 0), the airplane will trim at an angle of attack equal to about 66° and yaw divergence will occur.

To check the convergence, the initial starting values for the design variable are slightly changed. However, the final converged results are still the same, showing that the solution is the best possible one.

CONCLUSIONS

A numerical optimizer was coupled with a nonlinear flight dynamic analysis code to form a control system design method. For departure prevention, the design objective was described in terms of minimizing the transient angle of attack, the sideslip, the yaw angle, and the yaw rate, in a roll maneuver. To make sure the converged solutions were the desirable ones, the starting values of the system gains were obtained through a sensitivity analysis. The design method was demonstrated for an F-16 configuration to prevent pitch departure.

REFERENCES

1. Gilbert, W. P.; Nguyen, L. T.; and Van Gunst, R. W. "Simulator Study of Applications of Automatic Departure-and-Spin-Prevention Concepts to Variable-Sweep Fighter Airplane." NASA TM X-2928, 1973.
2. Nguyen, L. T.; Gilbert, W. P.; and Van Gunst, R. W. "Simulator Study of the Departure Resistance of Lightweight Fighter Airplane with Twin Vertical Tails." NASA TM X-3510, 1977.
3. Gilbert, W. P.; Nguyen, L. T.; and Van Gunst, R. W. "Simulator Study of the Effectiveness of an Automatic Control System Designed to Improve the High-Angle-of-Attack Characteristics of a Fighter Airplane." NASA TND-8176, 1976.
4. Gilbert, W. P.; and Libbey, C. E. "Investigation of an Automatic Spin Prevention System for Fighter Airplanes." NASA TND-6670, 1972.
5. Nguyen, L. T.; Gilbert, W. P.; and Ogburn, M. E. "Control System Techniques for Improved Departure/Spin Resistance for Fighter Aircraft." NASA TP-1689, 1980.
6. Fan, M. K. H.; Wang, L. S.; Koninckx, J.; and Tits, A. L. "CONSOLE: A CAD Tandem for Optimization-Based Design Interacting with User-Supplied Simulators." Workshop on Computational Aspects in the Control of Flexible Systems, NASA TM-101578, 1988, pp. 89-108.
7. Vanderplaats, G. N. "COPES/ADS--A FORTRAN Control Program for Engineering Synthesis Using the ADS Optimization Program." Engineering Design Optimization, Inc., June 1985.
8. Lan, C. E.; and Ge, F. "Application of Numerical Optimization Techniques to Control System Design for Nonlinear Dynamic Models of Aircraft." Proceedings of the Third Annual Conference on Aerospace Computational Control, JPL Publication 89-45, Vol. I, pp. 394-406.
9. Nguyen, L. T.; et al. "Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability." NASA TP-1538, December 1979.
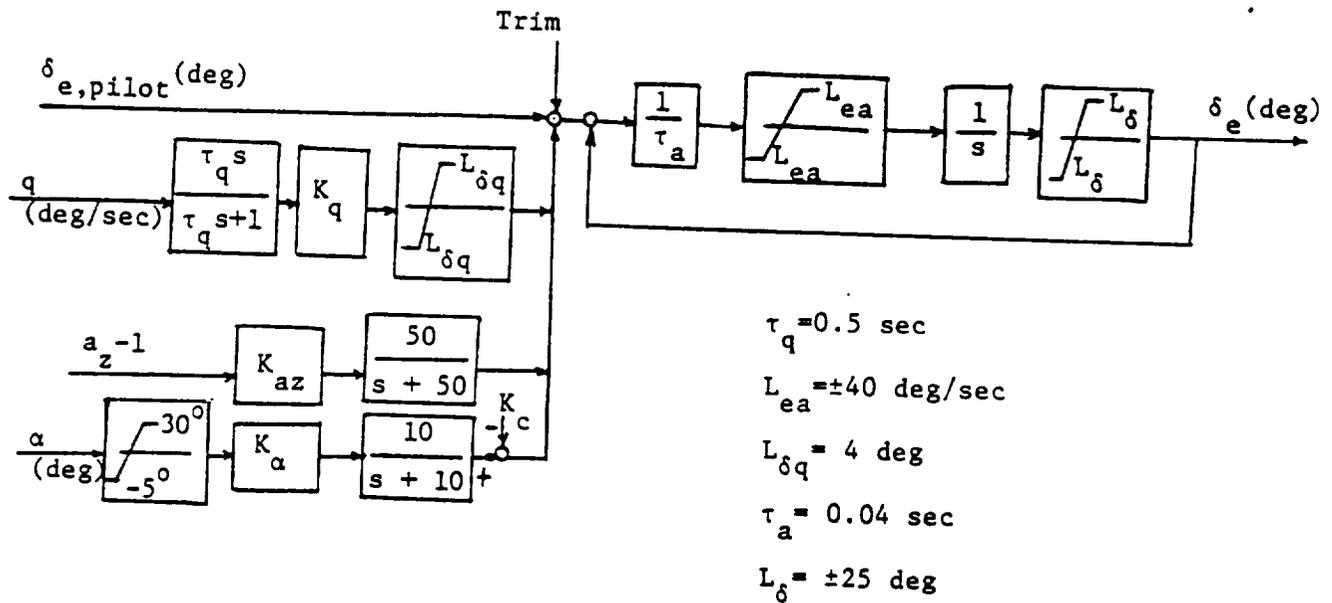
Figure 1 Block Diagram of an Assumed Pitch Control System for an F-16 Configuration

$\tau_q = 0.5$ sec

$L_{ea} = \pm 40$ deg/sec

$L_{\delta q} = 4$ deg

$\tau_a = 0.04$ sec

$L_{\delta} = \pm 25$ deg
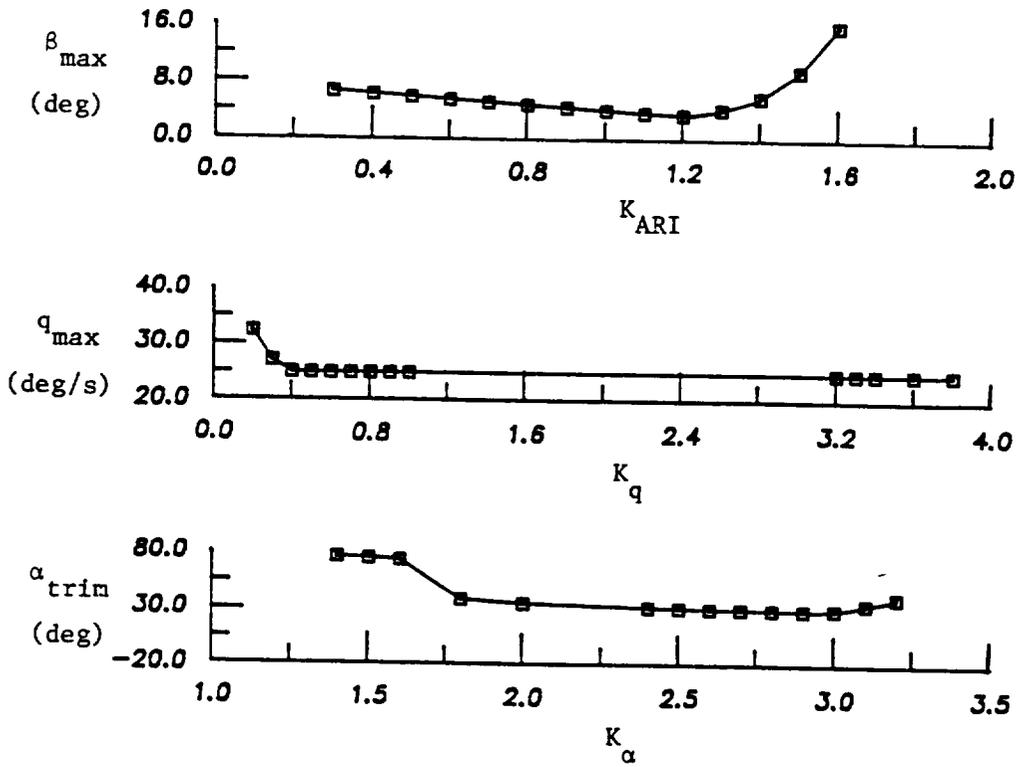
Figure 2 A Sensitivity Analysis of the Effect of System Gains on Dynamic Characteristics
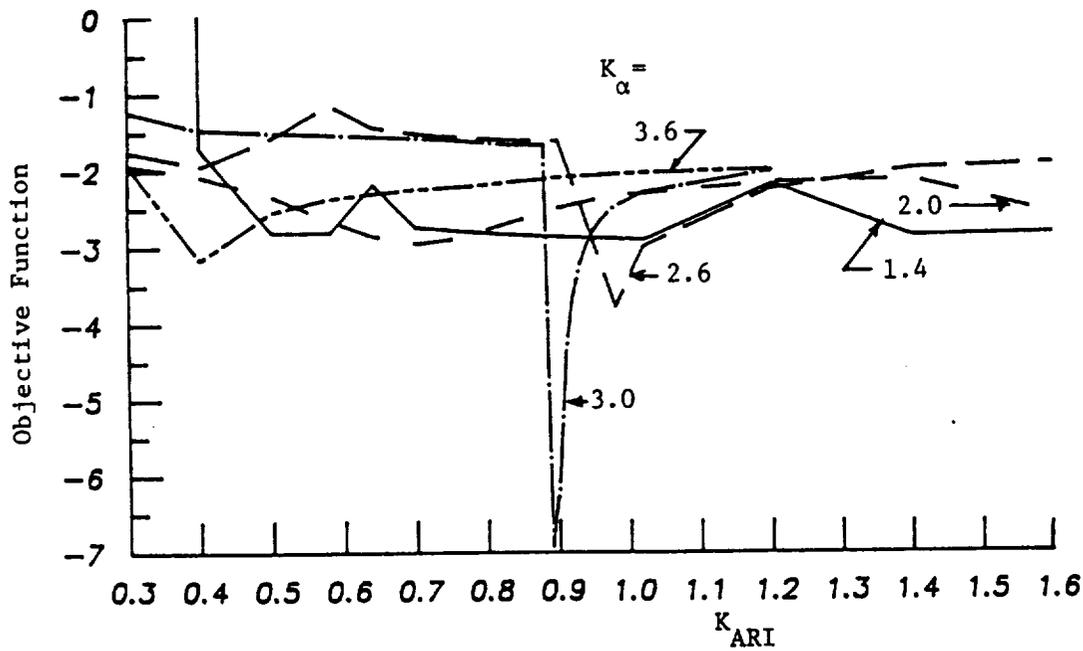
Figure 3   Variation of the Objective Function with K_{ARI}.

$K_{ay}=15$, $K_r=0.6$, $K_q=2.8$, $K_{az}=0.4$, $K_c=10$, $K_p=0.09$, $K_\delta=0.25$

K_{ARI}	K_{ay}	K_r	K_q	K_{az}	K_α	K_c	K_p	K_δ
0.77	10.	0.6	3.4	0.3	3.1	11.	0.09	0.25
0.77	10.	0.6	3.4	0.3	0.0	0.	0.09	0.25
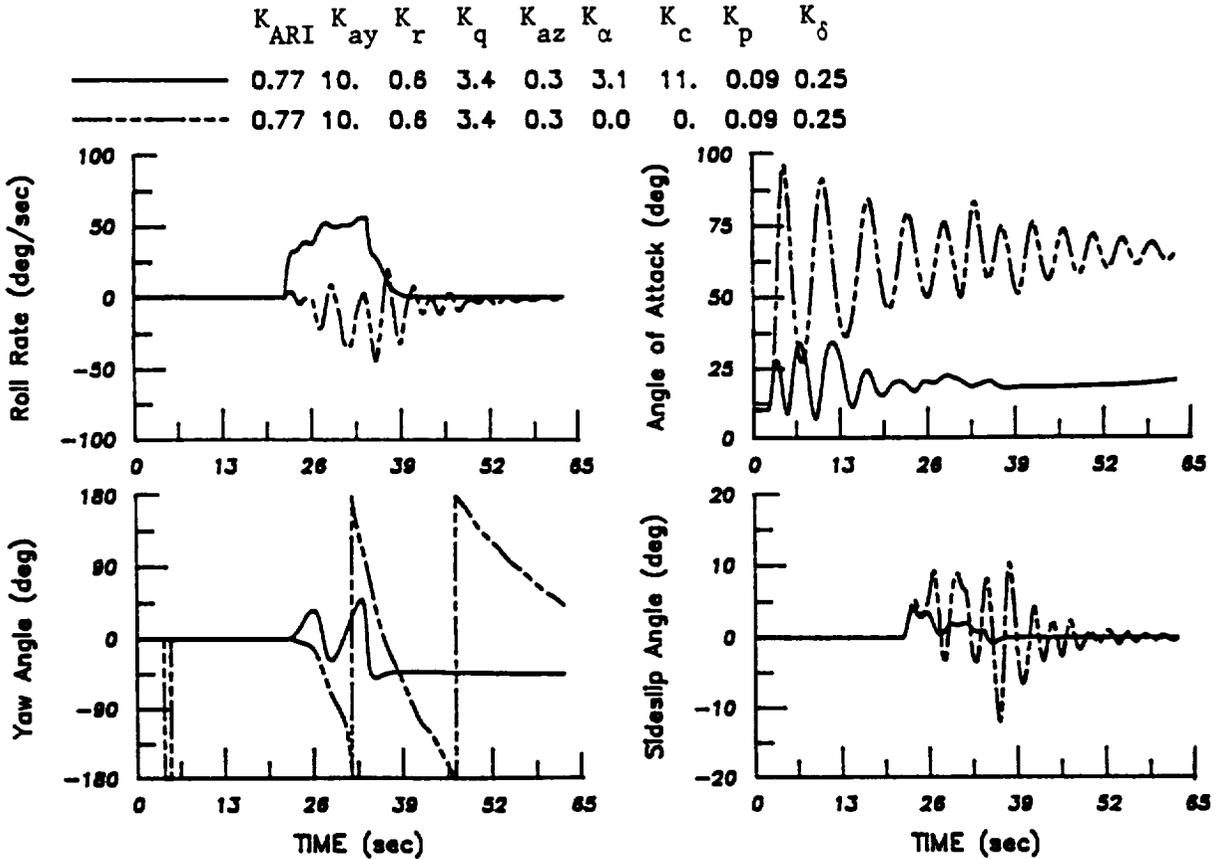
Figure 4   Design of a Control System to Prevent Pitch Departure of
an F-16 Configuration in a Pull-up and Roll Maneuver at
M = 0.5 and h = 30,000 ft.

568

ACTIVE FLUTTER AND GUST RESPONSE CONTROL

MULUNEH OLI
Laboratoire d'Aeronautique,Université de Liége,Belgique

After a short recall concerning the aeroelastic
equations, active control law based on optimal
stochastic control theory is synthesized for
a wing flutter and gust response. Robustness of
of the control system due to structured and un-
structured uncertainities is considered.Robust-
ness recovery technique is applied ·to improve
the stability margin.

INTRODUCTION

In the last three decades enormous progress has been made in the
field of structural analysis and numerical aerodynamics. This
has allowed to build yet lighter and more flexible aircrafts,
but the flutter phenomenon continues to be one of the decisive
aircraft performance limiting factors. The concern of this
paper is to study the control of flutter and turbelence response
to enhance the performance limitations.

A sustained oscillation of the wing in motion involving its
vibrational deformation due to the positive work done per cycle
of oscillation by the aerodynamic force leads to the flutter
phenomenon. To make sure that the phenomenon occurs out of the
flight envelope, active control system is being used. The
physical purpose of the control system is to generate
aerodynamic forces using of control surfaces to compensate the
adverse acting aerodynamic forces on the wing to keep the
aircraft stable and reduce load. In establishing the control
system stochastic optimal control design process is used. The
task of the control system is to find the control variables that
would drive the system from a state-space point in a Euclidean
space at the initial time to a point at the final time along an
optimal path.

The design model used to derive the control strategy, which is a
simplified version of the actual model, is obtained by
neglecting some of the modes and fixing some of the parameters
at nominal values. Robustness of the control system due to low
and high order uncertainities will be determined and a recovery
technique will be applied.

AEROELASTIC EQUATIONS OF MOTION

The elastic wing is considered to be linear which can be
discretised as n points material; so can its equations of motion
be formulated using Lagrange's equation of motion. The deformed
shape of the wing is represented by a set of discrete

displacements at selected nodes which is approximated by a linear combination of the natural modes of vibration.

$$\overline{q} = \Phi \, q \tag{1}$$

where $\overline{q}$ is the generalized displacement vector, Φ is the modal shape matrix and q is the modal coordinates vector.

$$M \, \ddot{q} + D \, \dot{q} + K \, q = F \tag{2}$$

The modal shape , mass and stiffness matrices Φ, M and K respectively are generated using the finit element method. The nonconservative generalized force matrix F consists of unsteady aerodynamic forces due to motion Q_m, control Q_c and atmospheric turbelence Q_t. The Laplace transform of the aeroelastic system equations of motion (2) is

$$(M \, s^2 + D \, s + K) \, q = \tfrac{1}{2} \rho \, v^2 Q_m q + \tfrac{1}{2} \rho \, v^2 Q_c \delta + \tfrac{1}{2} \rho \, v^2 Q_t \frac{wt}{v} \tag{3}$$

where δ is the control surface coordinates vector and wt is the atmospheric turbelence vertical speed.

The unsteady aerodynamic generalized forces are detemined using the subsonic doublet lattice method as follows. The lifting pressure – downwash integral equation (4):

$$w(x,y) = \frac{1}{4\pi} \iint_S \Delta P \, (x,y) \, K \, (x,x_1;y,y_1) \, ds \tag{4}$$

derived from the hyperbolic acceleration potential equation of a subsonic flow for a thin finite wing is discretized into linear algebraic equations

$$w = D \, \Delta P \tag{5}$$

where D is the influence coefficient matrix representing the downwash at a point (x,y) due to a unit pressure difference ΔP at (x_1,y_1). The downwash distribution is related to the wing modal coordinate vector and the pressure coefficients are evaluated using the corresponding modal deflections. The generalized unsteady aerodynamic forces associated with each generalized coordinate is

$$Q_{ij}(ik) = - \left(\frac{1}{b^2} \right) \iint_S q_i(x,y) \, \Delta p_j(x,y) \, dx \, dy \tag{7}$$

where s=ik ; k is the reduced frequency.

To cast equation (3) into the state space form , convenient for the control design, the unsteady aerodynamic force matrix obtained in the frequency domain is approximated by a rational matrix polynomial [7]

$$\hat{Q} \, (s) = A_0 + A_1 \left(\frac{cs}{2v} \right) + A_2 \left(\frac{cs}{2v} \right)^2 + \sum_{m=1}^{l} \frac{A_{m+2} \, s}{s + 2v/c \, \beta_m} \tag{8}$$

The coefficient matrices A_i are determined using the least squares technique.

CONTROLLER DESIGN

A state space equation has been constructed including flexible modes , aerodynamic lag modes , dynamic gust represented by the second order Dryden model and actuator dynamics.

$$\dot{X} = A \, X + B \, U + G \, W \tag{9}$$

$$Y = C \, X \tag{10}$$

where $X \in R^n$ is a state vector, $U \in R^m$ is a control vector, $W \in R^m$ is a turbelence vector and $Y \in R^l$ is a measuremert vector. The optimal control that minimizes a qudratic performance index (1 1)

$$J = E \int_0^\infty (X^T Q X + U^T R U) dt \qquad (11)$$

subject to (9) is obtained using the linear quadratic Gaussian (LQG) technique. The weighting matrices $Q \in R^{n \times n}$ and $R \in R^{m \times m}$ are so chosen to influence the required output performance and control cost respectively. The required outputs in-here are the flutter speed and the turbelence response.

To make the controller of less complexity a controller order reduction technique is to be applied. The modal cost method [3] is used in here for this purpose. It is used to make a modal ranking based upon relative contributions to a stochastic functional (11). The reduction is made after applying the optimal control technique. The system is assumed to be observable so as to assure the stability of the controller.

MODEL UNCERTAINITIES AND ROBUSTNESS

One of the principal objectives of using feedback control is to compensate for model uncertainities . Model uncertainities stem either from neglected dynamics or parameter uncertainities. To find the robustness of a control system with respect to unstructured uncertainities additive or multiplicative perturbations are introduced at different loop breaking points, since a robustness at a point in a multi-input multi-output (MIMO) system does not necessarily result at other points. Unstructured uncertainities in here are due to ignored higher wing structural modes, unsteady aerodynamics approximations and reduced order controller, whereas structured uncertainities are principally due to the natural frequecies and dynamic pressure. In terms of the loop transfer function the uncertainities are given as follows

$$G(s) = G_n(s) + \Delta G(s)$$

$$G(s,p) = G_n(s,p) + G(s,\Delta p) \qquad (13)$$

$$(12)$$

where G_n is nominal loop gain, p is a vecteur of uncertain parameters.

Robustness of a MIMO system with respect to unstructured uncertainities is evaluated in terms of the minimum singular value of the system's return difference matrix , which can be interprated as the distance between the return difference matrix and the nearest singular matrix or physically as the largest perturbation which can be tolerated at a point in a loop. The stability of a perturbed system is guaranteed [6] if

$$\underline{\sigma} (I + T) > \bar{\sigma} (L^{-1} - I) \qquad \forall \omega > 0 \qquad (14)$$

where T is the nominal loop gain and L is the perturbation matrix to account for the uncertainity in the open loop system. As the perturbation matrices cannot be exactly determined the perturbation norm bounds are used for the analysis. The bound on the perturbation matrix assumes a single worst uncertainity magnitude applicable to all channels.

ROBUSTNESS RECOVERY TECHNIQUE

Assuming that all state variables are accesible, linear quadratic regulators have impressive robustness properties [8], ie. phase margin of ±60° and gain margin = 1/2 ,∞ with a 50% reduction tolerance. The introduction of observers (filters) to estimate the unaccessible state variables deteriorates the robustness of the control system. Minimum phase plant's robustness which is not the case here ,with respect to model uncertainity can be asymptotically recovered at different points of the loop using the LQG/LTR [8] technique. Yedavali & Skelton using the trajectory sensitivity approach and Hyland & Bernstein using the entropy method have treated robustness with respect to parameter variation. The internal feedback loop technique of Tahk & Speyer is here used for this purpose. The parameter variation matrix ΔA is decomposed in to input, output and fictious feedback loop. An explicit relationship between the LQG weighting matrices and the the structure of the parameter variation is established. The regulator or the filter part of the LQG controller is asymptotically robustified with respect to the parameter variations udjusting the weighting matrices.
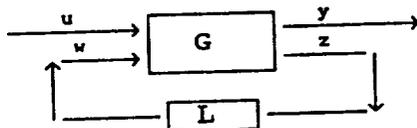
Fig. 1 Representation of the perturbation system

$$\Delta\ A(p)\ =\ -M\ L(p)\ N \qquad (15)$$

where p is the vecteur of uncertain parameters, $M \in \mathbb{R}^{nxp}$, $N \in \mathbb{R}^{qxn}$. The covariance of the process noise $v_1 = \gamma^2 R\ R^T$ is chosen so that M is column similar to R ie. span (M) $\subset$ span{N} and (A,R,C) is minimum phase. As $\gamma \to \infty$ the stability robustness is determined through the regulator gain. The asymptotic procedure can be applied to both the regulator and the filter by a simultaneous proper choice of the covariance matrix and the weighting matrix in terms of the input/output perturbation matrices M & N respectively.

EXAMPLE AND DISCUSSION

The finit element wing structural model consists of rod and beam elements. It is known that the flutter phenomenon dominantly involves the lower modes, so are only three lower modes retained for the control analysis and synthesis. The aerodynamic model (fig. 2) consists of 140 boxes. Unsteady aerodynamic calculations are made at a flight condition of 0.7 mach , altitude of 10 km and 6 reduced frequencies. The aerodynamic force rational polynomial fits are given on fig. 3. The initial weighting matrices are determined using the inverse square of the maximum allowable states and control variables. The system has been simulated for different values of the weighting matrice coefficients . Displacement, velocity and acceleration sensors are used to measure the outputs. The open loop flutter speed of the system (v_f=260m/sec) is sensibly increased (20%) when the control system is activated. Open and closed loop power spectral

density speed response to the atmospheric turbelence of scale, L=200m and gust speed intensity, vb=3.6m/sec,is given fig.4. One can conclude that the active flutter suppression and gust response control systems are related to one another concerning the flexible modes.

The minimum singular value curve of the return difference matrix at the input is used to determine the minimum acceptable perturbation matrix bound (worst case in all the channels) up to which stability is guaranteed (14). The guaranteed stability margins; PM = ±5° and GM = -1,1 db are determined from the multiloop gain and phase margins [6].

The principal necessary condition to apply Tahk's method is the minimum phasedness of (A,M,C), which is found to be varying with sensor location. When the uncertain parameters belong either to the control matrix B, or the measurement matrix C, the efficiency of this method gets reduced since it leads to a controller order augumentation, particularly for a system of higher inputs and outputs.

REFERENCES

1. J.K.Mahesh, C.R. Stone, W.L.Garrard & H.J.Dunn "Control law synthesis for flutter suppression using linear quadratic gaussian theory" J.Guidance & control vol.4 No 4 July-August 1981

2. B.S.Liebst, W.L.Garrard & M.Adams "Design of an active flutter suppression system" J.Guidance & Control, Jan-Feb. 1986.

3. R.E.Skelton & P.C.Hughes "Modal cost analysis for linear matrix second order systems", Journal of Dynamic systems, measurement, & control, vol. 102 sep. 1980.

4. D.C.Hyland & D.S.Bernstein, "The optimal projection equations for fixed order dynamic compensation", IEEE trans. on Automatic control, vol. AC-29, No.11, Nov. 1984

5. M.Tahk & J.L.Speyer, "Modeling of Parameter variations and asymptotic LQG synthesis", IEEE Trans. on automatic control, vol.AC-32, No.9,sep.1987.

6. V.Mukhopadhyay & J.R.Newsom,"A multiloop system stability margin study using matrix singular values", J.Guidance,V.7 No.5,sept.-oct.1984.

7. Roger, Kenneth L., "Airplane math modelling methods for active control design", Structural aspects of active controls. AGARD-CP- 228, Aug. 1977.

8. J.C.Doyle & G.Stein, "Multivariable feedback design: concepts for a classical/modern synthesis", IEEE Trans. on aut.cont. vol. AC-26,No.1, Feb.1981.

9. H.Kwakernaak,"Robust control", Minicourse on robust control & H_∞ theory. 1990 Benelux meeting on systems & control,March 1990.

10. R.Freymann, "Contribution of DFVLR, Active control application for flutter suppression & gust load alleviation" dited by H.Forsching , GARTEUR TP, 1986.
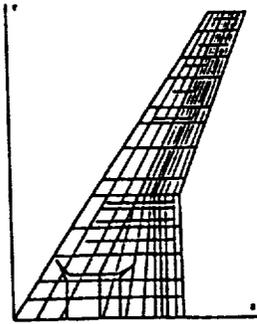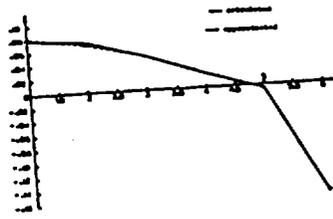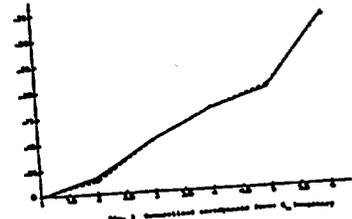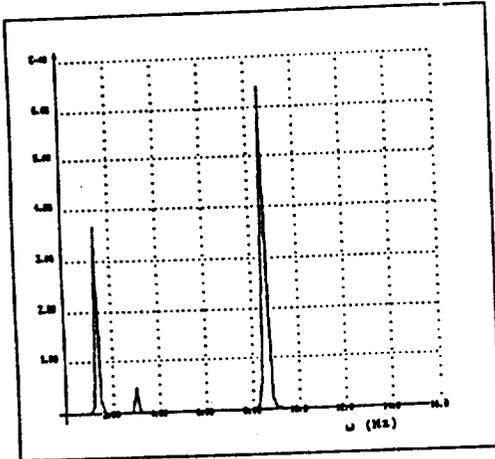
Fig. 4 controlled PSD speed response to AT
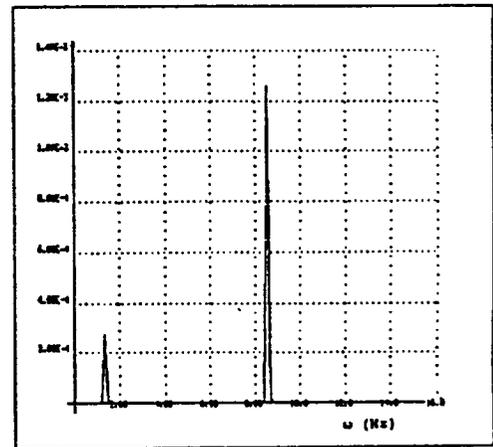
Fig. 4 uncontrolled PSD speed response to AT

3

$q_f = 4.5$ kpa

200

100

-20   -10   0   10

Real part, rad/sec

3

2

$q_f = 8.2$ kpa

Mode 1

200

100

-20   -10   0   10

Real part, rad/sec
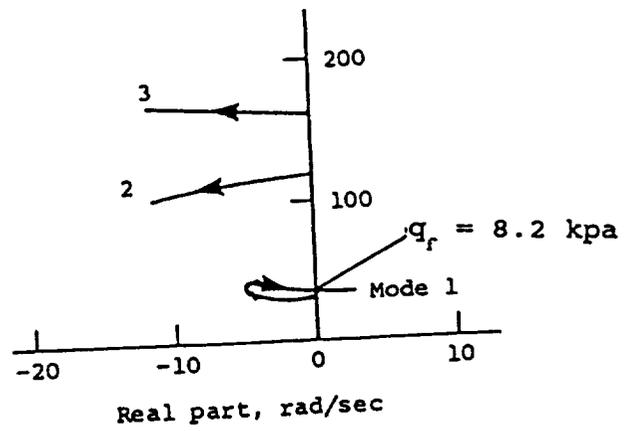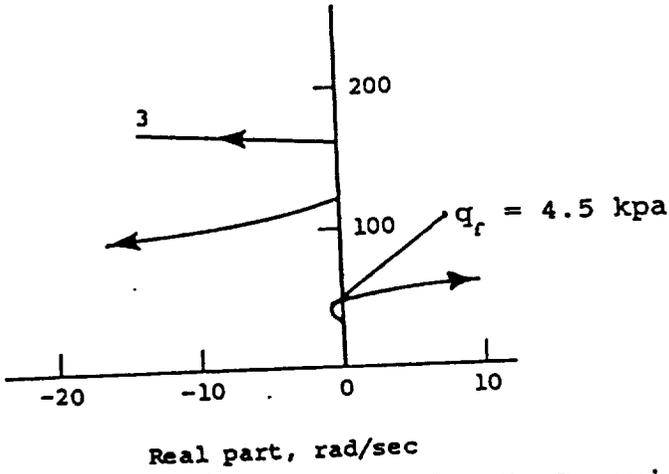
Fig. 7 Dynamic pressure root locus.
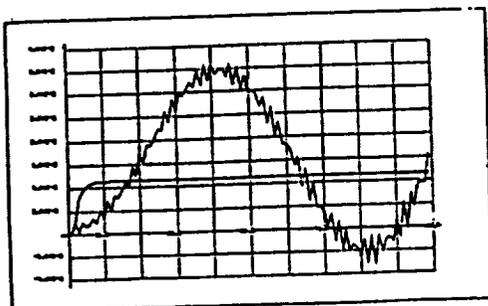
Fig. 6 Controlled and uncontrolled speed step response

6
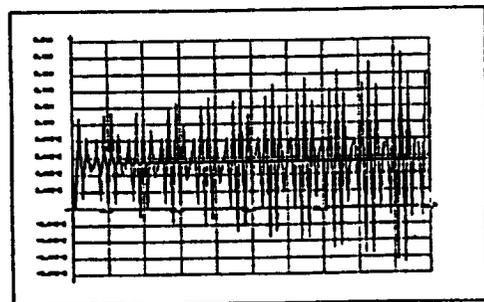
Fig. 8 Controlled and uncontrolled step displacement response

An Application of Object-Oriented Knowledge Representation to Engineering Expert Systems

by

D. S. Logie, H. Kamil and J. R. Umaretiya
Structural Analysis Technologies, Inc.
Santa Clara, CA

Abstract

The paper describes an object-oriented knowledge representation and its application to engineering expert systems. The object-oriented approach promotes efficient handling of the problem data by allowing knowledge to be encapsulated in objects and organized by defining relationships between the objects. An Object Representation Language (ORL) was implemented as a tool for building and manipulating the object base. Rule-based knowledge representation is then used to simulate engineering design reasoning. Using a common object base, very large expert systems can be developed, comprised of small, individually processed, rule sets. The integration of these two schemes makes it easier to develop practical engineering expert systems. The general approach to applying this technology to the domain of the finite element analysis, design and optimization of aerospace structures is discussed.

Introduction

An object-oriented approach is being applied to the development of an integrated Expert Engineering Software Package for structural analysis and design optimization of aerospace structures. The expert software package brings together several technologies, namely, expert systems, data base resources and procedural programs, to work together in an integrated environment as shown in Figure 1. Significant decision support is being built into the package which can be invoked by the user at various stages of the design process. Such an integration of the design process with embedded "intelligence" will improve the efficiency of the design process, help save significant man-hours and costs, and help produce reliable and economical designs.

The main goals of the research work presented herein were to:

☐ Develop a practical and efficient environment for the development of large expert systems, primarily for engineering applications, and

☐ Apply this environment to the aerospace design process involving the iterative steps of the finite element analysis, design and optimization.

After a preliminary review of the data and knowledge usually required for structural design, it was realized that the following two main capabilities were required:

❑ An efficient data representation and storage scheme, and

❑ The ability to process large numbers of rules necessary to simulate reasoning associated with the various steps of the design process.

It became clear that the object-oriented paradigm best provides a framework for defining the static problem data. An Object Representation Language (ORL) was therefore implemented in C++ as a tool for building, maintaining, and querying an object base. ORL commands, currently implemented, include those for creating, editing, and displaying classes and objects; querying, asserting, and reseting property values; and saving and loading objects. A main goal in the development was to keep the use of the ORL as simple as possible so that engineers or experts in other domains, without extensive computer programming experience, could develop knowledge bases and, furthermore, that non-experts could easily utilize the resulting expert systems.

Once the objects (data) involved and the relationships between the objects have been defined for a specific problem, the rules to reason with the object base can be developed. The CLIPS rule-based system was linked to the ORL to serve as the inference engine. Together, the object-oriented and rule-based schemes complement each other in that the object-oriented approach efficiently handles problem data while the rule-based knowledge is used to simulate the reasoning process. Alone, the object based knowledge is little more than an object-oriented data storage scheme; however, the CLIPS inference engine adds the mechanism to directly and automatically reason with that knowledge. In this hybrid scheme, the expert system dynamically queries for data and can modify the object base with complete access to all the functionality of the ORL from rules.

Object-Oriented Knowledge Representation

Object-oriented knowledge representation provides a natural means of representing problem data as a collection of related objects. Classes and objects are comprised of descriptive *properties* and *relationships.* A Property is a slot for holding values of type string, word, integer, or float. A Metaslot can be attached to a property to put constraints on the values the property can hold, to define initial or default values, and to define the prompt displayed to the user when queried. A powerful feature of a Metaslot is the "order of sources" which is a list defining the search path to be followed when the object property is queried. Currently, the User is the default source for information; however, "intelligent" interfaces to Data Base Management Systems (DBMS's) and Procedural Programs are being developed.

Relationships allow properties to be inherited by related classes and objects. Types of relationships include *instance* and *instance-of* relationships between a class and its instance, *is-a* and *subclass* relationships between classes (e.g., **Jet** *is-an* **Airplane**, **Airplane** has *subclass* **Jet**) and *part-of* and *subobject* relationships between objects (e.g., **wing-x** is *part-of* **airplane-y**, **airplane-y** has *subobject* **wing-x**). The relationships come into play when the classes and objects are queried. If a class is queried for a property value, it will automatically pass the query on to its instances. Similarly, if an object is queried for a property value which it doesn't have, it may pass the query on to related objects according to the current inheritance protocol. The relationship capability promotes efficient handling of data by eliminating unnecessary redundancy.

As illustrated in Figure 2, a powerful advantage of using an object-oriented knowledge representation scheme, independent of the rule-based representation and inference engine, is the ability to build a large expert system out of small rule sets that carry out specific tasks. Also, previously autonomous expert systems can share data and communicate through a common object base. A very large network of rule sets can be developed giving the illusion of a large expert system when, in fact, only a small set of rules are being processed at any one time. This capability becomes especially important on a personal or desktop computer platform. Developing, modifying, updating and verifying knowledge bases for large applications is a less formidable task when small rule sets can be edited and tested independent of the entire application.

Application to Aerospace Structural Design

A major objective of this effort has been to develop a knowledge base that mimics an expert in making key technical and conceptual decisions in the field of structural analysis and design optimization, thus improving the productivity with modern structural analysis/design optimization tools. The following advisory rule sets are operational and others are under development:

Analysis Planning
Substructuring
Mesh Refinement
Element Selection
Dynamic Modeling
Optimization Problem Formulation
Optimization Problem Simplification
Optimization Algorithm Selection
Interactive Design Optimization

A simplified version of the Analysis Planning Advisor will be used to illustrate the application of the object-oriented knowledge representation scheme to develop expert systems. The first step in developing the expert system is to define the objects and properties that influence the decision of what type of analysis to perform on the model. For the sake of this discussion, assume that the decision is solely based on the current goal of the design project and the loadings imposed on the structure. Objects for the expert system could be defined as follows:

```
OBJECT goal
   PROPERTY type TEXT
      METASLOT
         Allow (
               1. "Initial Design"
               2. "Final Design" )
         Prompt "What is the current goal of the Project? "

OBJECT structure
   PROPERTY loading TEXT
      METASLOT
         Allow (
               1. "Pressure"
               2. "Thermal"
               3. "Aeroelastic"
               4. "Aerodynamic"
               5. "Impactive" )
         Prompt "What loadings are imposed on the
   structure?"

OBJECT analysis
   PROPERTY type TEXT
      METASLOT
         Allow (
               1. "Static"
               2. "Dynamic - Time History"
               3. "Dynamic - Modal" )
```

Rules are then defined to reason with the objects. Rules query the objects for property values, automatically querying the user if their values are *unknown*. Other rules are required to display results or recommendations. Most importantly, rules must be established to mimic the experts reasoning. In this simple analysis advisor, a rule, shown in the CLIPS syntax, might look like:

```
(defrule rule-name
   (goal type "Final Design")
   (structure loading Aerodynamic)
=>
   (ORL assert analysis type "Dynamic - Time History") )
```

This simple, illustrative rule states that if the project is in the Final Design stage and the structure has an Aerodynamic loading then a Time History analysis should be

performed. The actual Analysis Planning Advisor has several more objects involved with many more possible values.

Conclusions

The paper described development of an object-oriented knowledge representation which was linked to the CLIPS rule-based system and is being applied to the aerospace structural design process. Expert systems of unlimited size can be developed by allowing managable sized rule sets to be chained together via a common object base. Because the object base is independent of the rule sets, the objects (data) can be shared among the rule sets. User responses and inferences of a rule set are stored for later use in other rule sets, completely transparent to the user. A simplified version of the Analysis Planning Advisor was used to illustrate how this knowledge representation scheme is being used to develop an "intelligent" software package for the design of aerospace structures.

Acknowledgements

References

1. Kamil, H., Vaish, A.K., and Berke, L., *An Expert System for Integrated Design of Aerospace Structures*, Fourth International Conference on Application of Artificial Intelligence in Engineering (AIENG 89), Cambridge, England, July, 1989.

2. Logie, D. S., Kamil, H., *Integration of Object-Oriented Knowledge Representation with the CLIPS Rule-Based Systems*, First CLIPS Users Group Conference, Houston, August, 1990.
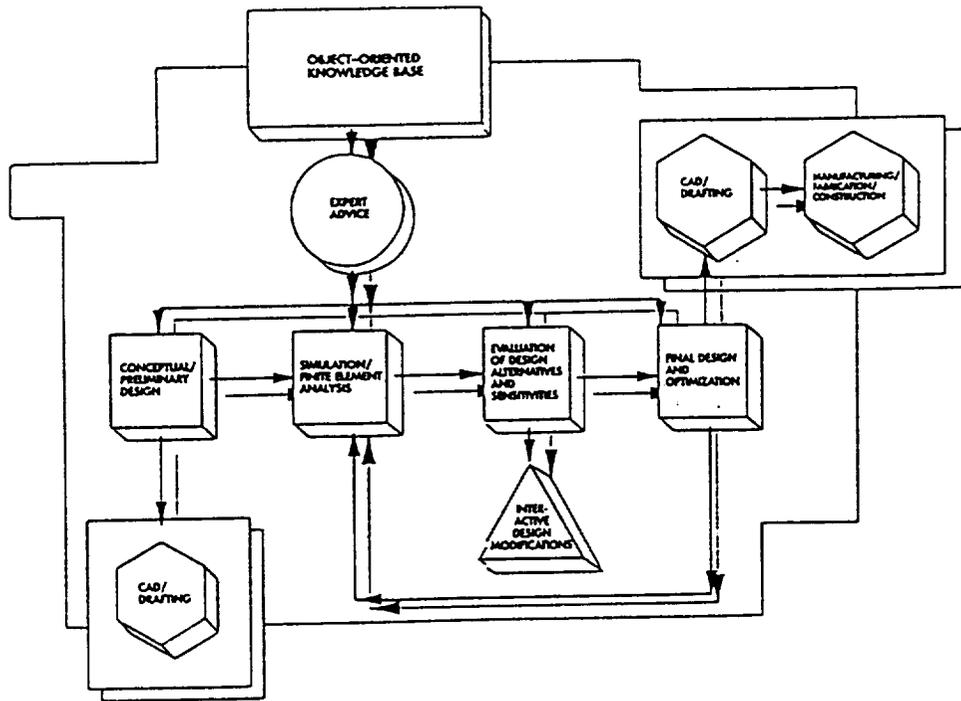
Figure 1. Integrated Architecture for Expert Design Software
Package Utilizing Object-Oriented Knowledge Representation
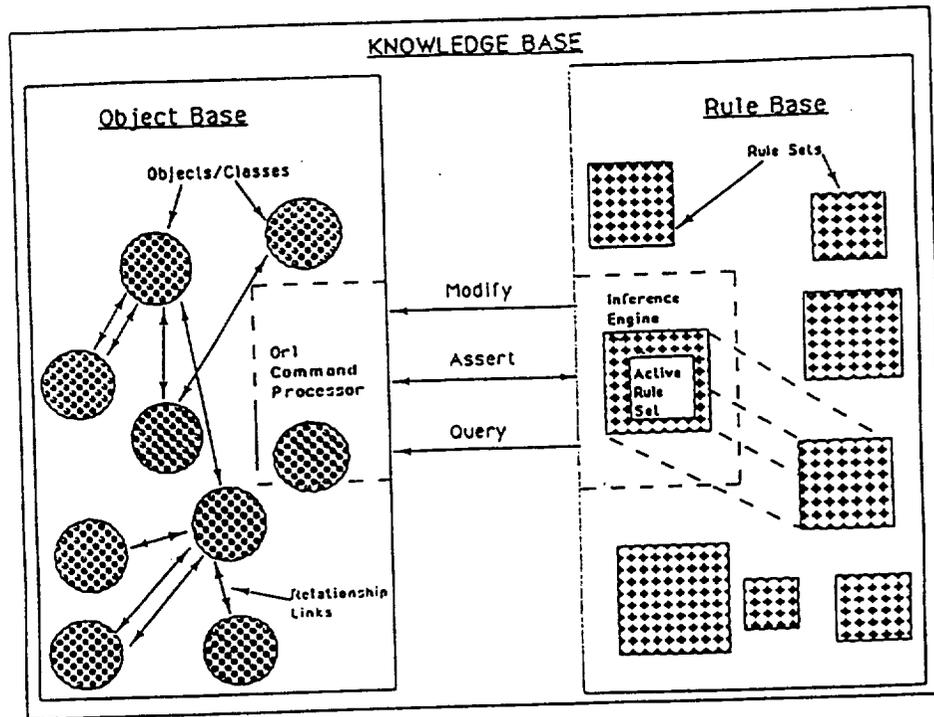
Figure 2. Integrated Object-Oriented/Rule-Based
Knowledge Representation Scheme

C-8.

Second-Order Design Sensitivities for Linear Elastic Problems

Qing Zhang and Subrata Mukherjee
Dept. of Theoretical and Applied Mechanis, Cornell University
Ithaca, NY14853

Abstract

The subject of this paper is the efficient and accurate determination of second-order design sensitivities in elastic bodies. The approach being carried out here is the direct differentiation of the governing derivative boundary element method (DBEM) formulation of the problem. Second-order sensitivities of boundary stresses are obtained here in an elegant manner. A numerical implementation of the method is carried out with isoparametric quadratic boundary elements and numerical results are presented for several sample problems. Considerable savings in computational effort for an optimization procedure is possible through the use of efficiently determined accurate second-order design sensitivities.

1. Introduction

In almost all shape optimization processes, design sensitivity coefficients(DSCs), which are the rates of mechanical quantities with respect to a design variable, are essential for the determination of the optimum shape of a body. A design variable being considered here is a shape parameter that controls the shape of part or whole of the boundary of a body. Approaches for calculating first-order DSCs have been developed quite well by many researchers. Second-order DSCs, however, are hardly used, mainly because they are very expensive to compute. In many nonlinear programming algorithms for optimization, second-order information such as the inverse of the Hessian matrix of the objective function and second-order derivatives of constraints, which provide sufficient conditions for the minimum (optimum) design, are generated approximately from first-order derivatives [1]. The quasi-Newton algorithm is popular for this purpose. However, another side of the issue is: if second-order DSCs can be determined accurately with reasonable computing cost, great increase of the rate of convergence of some optimization algorithms is possible [2]. This can lead to great savings in overall computing costs.

Several researchers have alreadly shown interest in this area. An adjoint variable method was used to derive first-order and second-order derivatives of measures of dynamic response with respect to design variables by Haug and Ehle [3]. Dems used the mixed approach to obtain second-order DSCs for conduction systems [4, 5]. Haftka compared two commonly used first-order approximations of the constraints to the corresponding second-order approximations of the constraints. In this paper [6] truss and laminated plate problems were used to compare the accuracy of the approximation and its effect on computational efficiency. Besides, Haug et.al [7] described the general approaches by using finite element methods (FEM). Direct differentiation of the reduced global stiffness matrix, the adjoint variable approach with the reduced global stiffness matrix, and a combination of direct differentiation and adjoint methods, have been used to formulate second-order design sensitivities [7]. None of the above authors, however, have attempted to determine second order DSCs for general elastic continuua.

The approach being used here is the direct analytical differentiations (DDA) of the governing boundary element method (BEM) formulation of the problem. The exact differentiations eliminate errors that might occur from finite difference methods and lead to closed form integral equations for the desired second-order derivatives. These equations are then solved by numerical discretization. This approach is accurate and efficient.[8,9].

2. Integral equations for two-dimensional problems

2.1 The DBEM formulation

A derivative boundary element method(DBEM) formulation for (two-dimensional) linear elasticity, in which the tractions and tangential derivatives of displacements are the primary variables on the boundary of a body, has been proposed by Ghosh et al [10]. An analogous formulation has been presented also by Okada, Rajiyah and Atluri [11].

The DBEM equations for two-dimensional linear elasticity for a simply connected region B can be written as [10]:

$$\int_{\partial B} [U_{ij}(P,Q)\, \tau_i(Q) - W_{ij}(P,Q)\, \Delta_i(Q)]\, ds(Q) = 0 \qquad (1)$$

where the kernel U_{ij} is available in many references eg.[12] and W_{ij} is available in [10]. Here τ_i and Δ_i are the components of the traction and tangential derivative of the displacement, $(\Delta_i = \partial u_i/\partial s)$ respectively, on ∂B. It is very important to note that W_{ij} has only a logarithmic singularity (same as U_{ij}) as r goes to zero. A constraint equation

$$\int_{\partial B_1} \Delta_i\, ds = u_i(2) - u_i(1)$$

(where ∂B_1 is a suitable part of ∂B with $u_i(1)$ and $u_i(2)$ the values of u_i at the beginning and at the end of ∂B_1) must be included for certain problems.

As can be seen from equation (1), the traction and tangential displacement derivative vectors are the primary unknowns on ∂B in this formulation. It has been shown that the stress components at a regular point on ∂B can be written in terms of the components of τ and Δ as [13]:

$$\sigma_{ij} = A_{ijk}\, \tau_k + B_{ijk}\, \Delta_k \qquad (2)$$

where A_{ijk} and B_{ijk} are coefficients which can be determined from the geometry and material properties [13]. Thus, since τ and Δ are primitive variables on ∂B in this DBEM formulation, these quantities, as well as σ_{ij}, can be obtained on ∂B with very high accuracy.

2.2 First-order sensitivity formulation

The corresponding DBEM equation for the sensitivities are obtained by differentiating equation (1) with respect to a shape design variable b [8]:

$$\int_{\partial B} [U_{ij}(b,P,Q)\, \overset{*}{\tau}_i(b,Q) - W_{ij}(b,P,Q)\, \overset{*}{\Delta}_i(b,Q)]\, ds(b,Q)$$

$$+ \int_{\partial B} [\overset{*}{U}_{ij}(b,P,Q)\, \tau_i(b,Q) - \overset{*}{W}_{ij}(b,P,Q)\, \Delta_i(b,Q)]\, ds(b,Q)$$

$$+ \int_{\partial B} [U_{ij}(b,P,Q)\, \tau_i(b,Q) - W_{ij}(b,P,Q)\, \Delta_i(b,Q)]\, \overset{*}{d}s(b,Q) = 0 \qquad (3)$$

where a superscribed * denotes a first-order derivative with respect to a typical component of b. It has been shown [14] that

$$\overset{*}{U}_{ij}(b,P,Q) = U_{ij,k}(b,P,Q)\, [\overset{*}{x}_k(Q) - \overset{*}{x}_k(P)] \qquad (4)$$

where, by virtue of the fact that

$$\overset{*}{x}_k(Q) - \overset{*}{x}_k(P) \sim O(r)$$

U_{ij} is completely regular! A similar argument is used to show that W_{ij} is also regular. The formula for $\overset{*}{d}s$ is given in [8,9].

2.3 Second-order sensitivity formulation

The corresponding DBEM equation for the second-order sensitivities is developed by extending the above idea. Differentiating equation (3) with respect to a shape design variable b, one can obtain:

$$\int_{\partial B} [U_{ij}(b,P,Q)\, \overset{\triangle}{\tau}_i(b,Q) - W_{ij}(b,P,Q)\, \overset{\triangle}{\Delta}_i(b,Q)]\, ds(b,Q)$$

$$+ 2\int_{\partial B} [\overset{*}{U}_{ij}(b,P,Q)\, \overset{*}{\tau}_i(b,Q) - \overset{*}{W}_{ij}(b,P,Q)\, \overset{*}{\Delta}_i(b,Q)]\, ds(b,Q)$$

$$+ 2\int_{\partial B} [U_{ij}(b,P,Q)\, \overset{*}{\tau}_i(b,Q) - W_{ij}(b,P,Q)\, \overset{*}{\Delta}_i(b,Q)]\, d\overset{*}{s}(b,Q)$$

$$+ 2\int_{\partial B} [\overset{*}{U}_{ij}(b,P,Q)\, \tau_i(b,Q) - \overset{*}{W}_{ij}(b,P,Q)\, \Delta_i(b,Q)]\, d\overset{*}{s}(b,Q)$$

$$+ \int_{\partial B} [\overset{\triangle}{U}_{ij}(b,P,Q)\, \tau_i(b,Q) - \overset{\triangle}{W}_{ij}(b,P,Q)\, \Delta_i(b,Q)]\, ds(b,Q)$$

$$+ \int_{\partial B} [U_{ij}(b,P,Q)\, \tau_i(b,Q) - W_{ij}(b,P,Q)\, \Delta_i(b,Q)]\, d^{\triangle}s(b,Q) = 0 \qquad (5)$$

where a superscribed $\triangle$ denotes a second-order derivative with respect to a typical component of **b**. $\overset{\triangle}{U}_{ij}$ is regular since $\overset{*}{U}_{ij}$ is already regular. A similar argument is used to show that $\overset{\triangle}{W}_{ij}$ is also regular.

The first lines of equations (3) and (5) are identical to that of equation (1) with either the first or the second-order sensitivities replacing the tractions and displacement derivatives. Half of sensitivities on ∂B must be prescribed and the rest can then be determined from equations (3) and (5). The known right hand side (when solving equation (3), τ_i and Δ_i on the ∂B are known and when solving equation (5) τ_i, Δ_i, $\overset{*}{\tau}_i$ and $\overset{*}{\Delta}_i$ on ∂B are known) involves the evaluation of regular integrals which is very easy to perform accurately.

The first and second order sensitivity equations for boundary stresses, obtained by differentiating equation(2) with respect to b, are given in [9].

Careful attention is paid in this work to the matter of modelling of corners(on ∂B) with conforming elements. Corners usually have jumps in τ and Δ. The stress σ may or may not be continuous at a corner. In the example presented in this work, the stress components are continuous at corners. A detailed discussion of this issure, including extra equations at corners, is available in[8,9].

3. Numerical implementation

3.1 Discretization of equations

The BEM equations (1) (for tractions and tangential displacement derivatives), equation (3) (for their first-order sensitivities) and equation (7) (for their second-order sensitivities) are discretized in the usual way. The boundary ∂B is subdivided into piecewise quadratic, conforming boundary elements. The variables τ_i and Δ_i are assumed to be piecewise quadratic on these boundary elements. The logarithmically singular kernels are integrated by using *log*-weighted Gaussian integration.

583

3.2. Numerical Results

Two sample problems are solved by the current method. These examples are chosen with a view of comparing DBEM numerical results with analytical solutions. All the numerical results discussed below are with Poisson's ratio $\upsilon = 0.3$. The mechanical quantities τ, Δ and σ and their first- and second-order sensitivities are determined for each problem. Results for only the second-order sensitivities are presented in the current paper. The first is plane stress problem. The second one is plane strain. Figures 1-4 and Table 1 are taken from [9].

Example 1 . The problem of a disk with external pressure (plane stress) is considered in this example. Only a quarter of the disk needs to be modelled because of symmetry (figure 1). Here, a=4, b=10, p = 1.0. The corners here arise due to the use of symmetry of the problem. They are corners where the stresses are continuous. The inner radius a is the design variable in this problem. The well known analytical solution for stress components in polar coordinates are given in [15]:

The analytical expression for second-order sensitivities of the stress components are not listed here since they are very long.

A comparisons of analytical and numerical results for second-order stress sensitivities (on the line DC in figure 1) is shown in figure 2 The sensitivities are in consistent units. A total of 44 quadratic elements (11 elements are equally spaced on each segment) are used to obtain the numerical results. In this figure, the smooth curve is the analytical solution. The numerical result, except for some small oscillations in some cases, agrees well with the analytical solution over the entire region.

A convergence study for this problem, with different meshes, has been carried out. Table 1 shows the results for σ^Δ_{11} at point C^+ (figure 1) from different meshes. The number of quadratic elements n in the table are for the whole boundary (a quarter of the disk). Each of the four segments has n/4 elements. On a given segment, the elements are equally spaced. It should be noted that while the results from the finer meshes are very good, convergence is not monotonic in this problem.

Example 2. The classical problem of a body with an elliptical hole is considered in this example. Only a quarter of the ellipse needs to be modelled because of symmetry (figure 3). Here, a=2, b=1, L=30, $\sigma_\infty = 1.0$. As in the previous example, this problem also has corners where the stress are continuous. The semi-major axis a is the design variable in this problem. The analytical solution for the tangential stress on the ellipse (for an ellitical hole in an infinite plate) is [14]:

$$\sigma_{\theta\theta} = \sigma_\infty \frac{1 + 2q - q^2 + 2\cos(2\phi)}{1 + q^2 + 2q\cos(2\phi)}$$

where ϕ is the eccentric angle and q=(b-a)/(b+a). The angle ϕ=0 at the point A in figure 3. Again, the analytical expression for second-order sensitivity of the tangential stress is not listed here since it is very long.

The mesh is exactly the same as that used in ref.[8]. A total of 54 quadratic elements (20 elements are spaced at equal increments of the eccentric angle on the quarter ellipse, 12 elements are used on AB, 14 on DE, 4 on BC and 4 on CD, respectively) are used for the numerical results. The density of elements on AB and ED is nonuniform, with small elements being placed near the points A and E, respectively. A comparison of analytical and numerical results for the second sensitivity of the tangential stress along the quarter ellipse is shown in figure 4. It is important to note that most authors typically only present results at the stress concentration point A [14] while the global picture is presented here. Also, it is important to bear in mind that these are second sensitivities which are generally acknowledged to be difficult to obtain accurately in an efficient manner.

References

[1] J.S. Arora, Introduction to Optimum Design, Class notes (The university of Iowa, Iowa city , 1988)

[2] C. Fluery, Efficient approximation concepts using second order information, Proc.AIAA/ ASMS/AHS 29th SDM conference (Blacksburgh, VA, 1988)

[3] E.J. Haug and P.E. Ehle, Second-order design sensitivity analysis of mechanical system dynamics, Internat. J. Numer. Methods Engng. 18 (1982) 1701-1717.

[4] K. Dems, Sensitivity analysis in thermal problems-I: Variation of material parameters within a fixed domain, J. Therm. Stresses, 9 (1986) 303-324.

[5] K. Dems, Sensitivity analysis in thermal problems-II: Structural shape variation, J.Therm. Stresses, 10 (1987) 1-16.

[6] R.T. Haftka, First and second order constraint approximations in structural optimization, Computational Mech. 3 (1988) 89-104.

[7] E.J. Haug, K.K. Choi and V. Komkov, Design Sensitivity Analysis of Structural Systems (Academic Press Inc. New York, 1986).

[8] Q. Zhang and S. Mukherjee, Design sensitivity coefficients for linear elastic bodies with zones and corners by the derivative boundary element method, Int. J. Solids Structures (in press).

[9] Q. Zhang and S. Mukherjee, Second-order design sensitivity analysis for linear elastic problems by the derivative boundary element method. Comp. Meth. Appl. Math. Engng. (in press).

[10] N.Ghosh, H. Rajiyah, S. Ghosh and S. Mukherjee, A new boundary element method formulation for linear elasticity, ASME J. of Appl. Mech. 53 (1986) 69-76.

[11]H. Okada, H. Rajiyah and S.N. Atluri, A novel displacement gradient boundary element method for elastic stress analysis with high accuracy. ASME J. Appl. Mech. 55 (1988) 786-794.

[12] S. Mukherjee, Boundary Element Methods in Creep and Fracture Elsevier Applied Science, London, (1982).

[13] J. Sladek and V. Sladek, Computation of stresses by BEM in 2-D Elastostatics, Acta Techanica CSAV. 31 (1986) 523-531.

[14] M.R. Barone and R.-J. Yang, Boundary intgral equations for recovery of design sensitivities in shape optimization, AIAA J. 26 (5) (1988) 589-594.

[15] S.P. Timoshenko and J.N. Goodier, Theory of Elasticity, Third edition. McGraw-Hill Inc. New York (1970).

Mesh	Analytical σ_{11}^{Δ}	Numerical σ_{11}^{Δ}	Error (%)
24 elements	-0.09342267	-0.09988122	6.46
28 elements	-0.09458990	-0.09988122	5.29
36 elements	-0.09816499	-0.09988122	1.72
40 elements	-0.1017195	-0.09988122	-1.84
44 elements	-0.1001325	-0.09988122	-0.25

Table 1. σ_{11}^{Δ} at point C^+ (in figure 1) from different meshes (example 1).
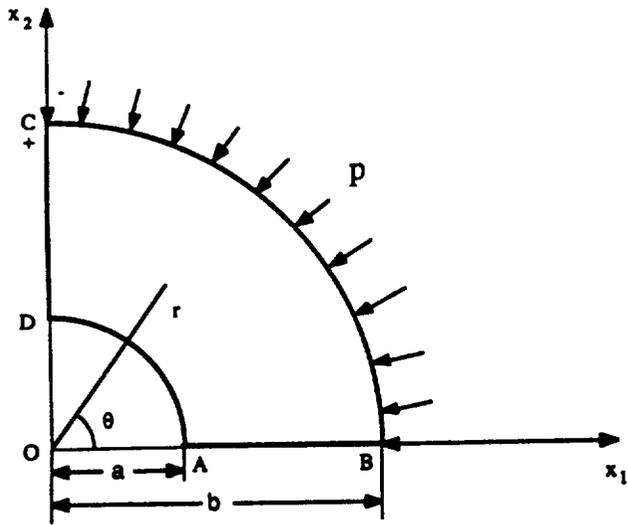
Figure 1. Example 1. Lame's problem for a thin disk.
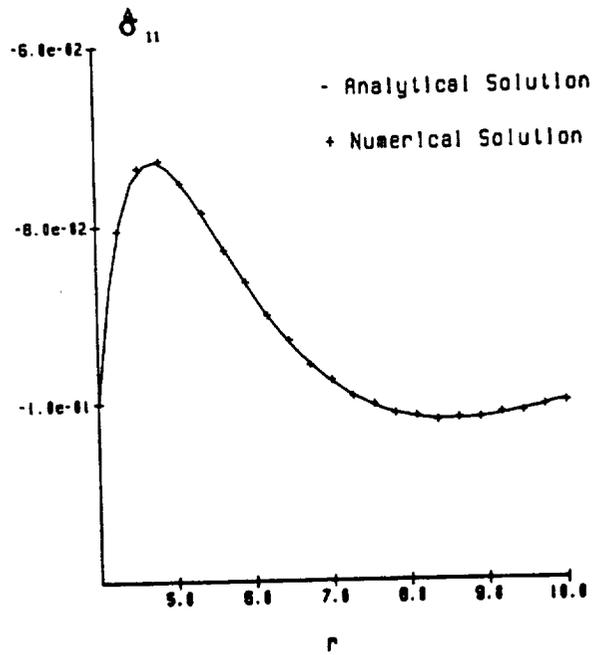
Figure 2. σ^{Δ}_{11} on DC--analytical and DBEM solution.
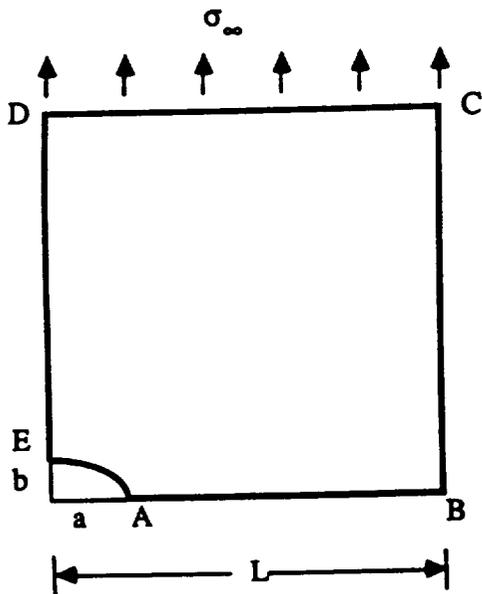
Figure 3. Example 2: A body with an elliptical hole.
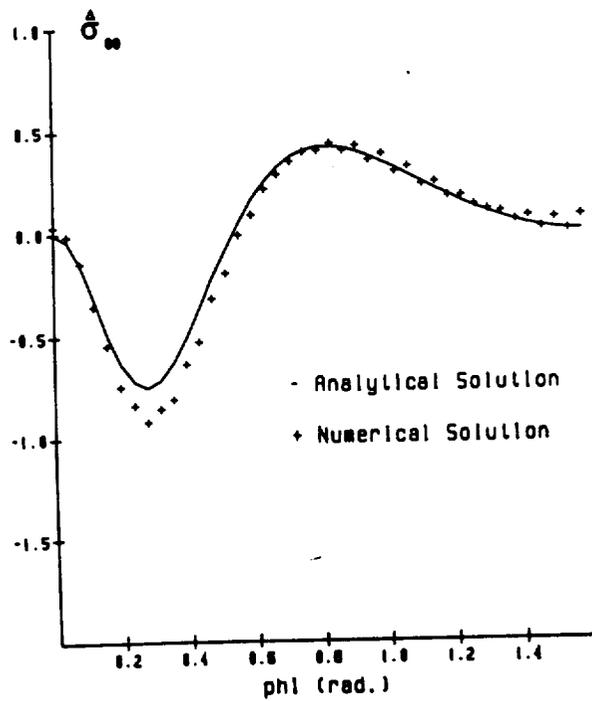
Figure 4. Angular variation of $\sigma^{\Delta}_{\theta\theta}$ around the quarter ellipse.

586

IMPLEMENTATION OF EFFICIENT SENSITIVITY ANALYSIS FOR OPTIMIZATION OF LARGE STRUCTURES

by

J. R. Umaretiya and H. Kamil

Structural Analysis Technologies, Inc.
Santa Clara, CA

ABSTRACT

The paper presents the theoretical bases and implementation techniques of sensitivity analyses for efficient structural optimization of large structures, based on finite element static and dynamic analysis methods. The sensitivity analyses have been implemented in conjunction with two methods for optimization, namely, the Mathematical Programming and Optimality Criteria methods. The paper discusses the implementation of the sensitivity analysis method into our in-house software package, AutoDesign™.

INTRODUCTION

The design process for aerospace structures involves many analysis/design iterations, exchanges of a large amount of data and multiple interactions on decisions among a variety of technical disciplines. Typically, the design process goes through several stages, ranging from early conceptual and preliminary designs, through finite element structural analyses, to final design and optimization. For each stage and cycle of the analysis and design, a large number of parameters are investigated and a large amount of data is utilized. The design process can readily be formulated as a problem of optimization, where either cost (weight) or a measure of performance can be optimized while satisfying the specified constraints.

The objective of our efforts have been to try to use optimization approaches for design of large, practical structures, based on the use of finite element techniques for modeling the structures. Large, practical structural optimization problems have hundreds or thousands of design variables, hundreds or thousands of highly nonlinear constraints and multiple local minima. For structural optimization, two different types of methods have been proposed, viz, Mathematical Programming (MP) methods (Ref. 2), and Optimality Criteria (OC) algorithms (Refs. 3,4). Both MP and OC methods are iterative in nature and generate a sequence

of design points in the space of the design variables converging to the optimum solution.

Both of these methods involve gradients of objective and constraint functions with respect to design variables. These gradients are called sensitivity coefficients. For regular optimization problems, these sensitivity coefficients can be evaluated directly since objective and constraint functions are explicit functions of design variables. Once these coefficients are known, they are used for determining the direction and size of the next iterative step towards optimum design. For structural optimization problems, the objective and constraint functions are implicit functions of the design variables. This makes the evaluation of their gradients significantly more difficult. The process of evaluating gradients of objective and constraint functions is called sensitivity analysis---which provides a bridge between optimization algorithms and finite element analysis solutions. This paper discusses the various approaches for sensitivity analyses and their implementation to large, practical structures.

SENSITIVITY ANALYSIS APPROACHES

There are three main approaches for sensitivity analyses[1]:

- Virtual Load Approach
- State Space Approach
- Design Space Approach

These methods are briefly discussed below:

Virtual Load Approach

This approach was first used by Barnett and Hermann (Ref. 5) for statically determinate trusses with a single displacement constraint. The approach was later extended to statically indeterminate structures with multiple deflection constraints, and has been extensively used with Optimality Criteria methods (Refs. 3,4).

Since this approach is somewhat restricted in the sense that constraints must be expressed in a specific form, it has been superseded by the more general

---

1 The finite difference approach is not considered herein, because of it's potential accuracy and reliability problems.

State Space and Design Space approaches, discussed below, from which the virtual load approach can be derived.

State Space Approach

In this approach, the state variable vector is first treated as an independent variable. An adjoint relationship is then introduced to express the effect of a variation in the state variable vector in terms of the variation in the design variable vector. In contrast to the virtual load approach, no special functional form of constraints is assumed and a variational approach is followed in deriving the design derivative vector.

Design Space Approach

This approach was first suggested by Fox (Ref. 6), and has been used by several researchers. In this approach, the state variable is not assumed to be independent, i.e., it is assumed to be a function of the design variable. This makes the equations more difficult to solve because of additional terms on the right hand side of the equations. The State Space approach is more efficient than the Design Space approach, especially for large, practical structures where the number of design variables is very large compared to the number of constraints.

Thus, it was decided to use the State Space approach for this research and development effort because of its generality and efficiency for large, practical structures.

IMPLEMENTATION OF SENSITIVITY ANALYSIS APPROACH

The State Space sensitivity analysis approach was implemented in our finite element analysis and design optimization software, AutoDesign™ (Ref. 1). This approach was implemented for both the MP and OC methods, for the truss, beam, membrane and plate/shell finite elements, for the displacement, stress and frequency constraints. Implementation for the 3-D solid (brick) elements and buckling constraints is in progress now. The implementation for the MP and OC methods is discussed below:

Both optimization approaches, namely MP and OC approaches, require the information on gradients of objective and constraint functions in addition

to information on functions themselves. As mentioned earlier, the constraint functions for structural design problems are usually functions of displacements, stresses, and frequencies, which are implicit functions of design variables. The gradients of constraint functions and objective function (generally weight or cost of the structure to be designed) are evaluated.

The objective function is usually given as:

$$W = \sum_{i=1}^{n} w_i A_i b_i$$

where w_i is the specific weight or cost

             A_i is the cross-sectional area of 1-D elements such as truss, beam, etc., or planar area of 2-D elements,

             b_i is the third dimension, namely, the length for 1-D elements or the thickness for 2-D elements,

and n is the number of elements in the finite element model of the given structure.

For 1-D elements, A_i's are used as the design variables, where as for 2-D elements, b_i's are used as the design variables. Thus, gradients of the objective function may be explicitly written as:

$$gW_i = \begin{array}{ll} w_i b_i & \text{for 1-D elements} \\ w_i A_i & \text{for 2-D elements} \end{array}$$

The computation of the gradients of constraint functions requires the evaluation of:

- partial differential coefficients of constraint functions with respect to design variables, say A

- partial differential coefficients of load vectors with respect to design variables, say B

590

- partial differential coefficients of stiffness matrix with respect to design variables, say $\underline{C}$

- partial differential coefficients of constraint functions with respect to displacements, say $\underline{D}$

- a solution vector, $\underline{\lambda}$, from $\underline{K}(b)\underline{\lambda} = \underline{D}$, where $\underline{K(b)}$ is the stiffness matrix

- partial differential coefficients of eigenvalues with respect to the design variables, say $\underline{E}$, (used for frequency and buckling constraints only)

For displacement constraints, $\underline{A} = 0, \quad \underline{E} = 0.$

For stress constraints, $\underline{E} = 0$

For frequency and buckling constraints, $\underline{A} = 0, \quad \underline{\lambda} = 0.$

In general, evaluation of $\underline{A}, \underline{B}, \underline{C},$ and $\underline{E}$ is straight forward since it can easily be performed at element level. The evaluation of $\underline{\lambda}$ requires special attention to make the sensitivity analysis efficient, since all the optimization algorithms require the information on constraints before they require the information on the gradients of the constraint functions. The evaluation of constraint functions requires the analysis of the finite element model of the given structural problem.

Now, if $\underline{D}$ is introduced as a set of dummy loads for each constraint, then $\underline{\lambda}$ can be obtained as a set of resulting dummy displacements from the analysis. Thus, for the optimization problem, the total number of load cases are considered to be the actual number of load cases plus the number of constraints in the design problem. Using this approach for multiple load cases, the stiffness matrix, $\underline{K}(b)$, is required to be decomposed only once, resulting in very efficient computation of $\underline{\lambda}$.

This approach was very successful in improving the efficiency of the sensitivity analyses implemented in our software AutoDesign™, especially for large, practical structures.

CONCLUSIONS

The sensitivity analysis approaches for structural optimization utilizing Mathematical Programming and Optimality Criteria methods were discussed. The implementation of sensitivity analysis into AutoDesign™, our in-house finite element analysis and design optimization software, was presented, especially considering application to large, practical structures. The State Space sensitivity analysis approach was utilized, with certain modifications, since it was found that this was the most suitable approach for large, practical applications considering its generality and efficiency.

REFERENCES

1. *User's Guide, AutoDesign™: An "Intelligent" Integrated Analysis and Design System for 3-D Structures,* June 1990.

2. Arora, J.S., and Haug, E.J., *Methods of Design Sensitivity Analysis in Structural Optimization,* AIAA Journal, Vol. 17, No. 9, 1979, pp 970-974.

3. Berke, L., and Khot, N.S., *Use of Optimality Criteria Methods for Large Scale Systems,* AGARD-LS-70, 1974, pp 1-1 to 1-29.

4. Venkayya, V.B., *Optimality Criteria: A Basis for Multi Disciplinary Design Optimization,* Computational Mechanics, 5, 1 - 21.

5. Barnett, R. L. and Hermann, P.C., *High Performance Structures,* NASA, CR - 1038, 1968.

6. Fox, R. L., *Constraint Surface Normals for Structural Synthesis Techniques,* AIAA Journal, Vol.3, August 1965.

Design Sensitivity Derivatives for Isoparametric Elements by Analytical and Semi-analytical Approaches

Kenneth W. Zumwalt and Mohamed E. M. El-Sayed

Abstract

This paper presents an analytical approach for incorporating design sensitivity calculations directly into the finite element analysis. The formulation depends on the implicit differentiation approach and requires few additional calculations to obtain the design sensitivity derivatives. In order to evaluate this approach, it is compared with the semi-analytical approach which is based on commonly used finite difference formulations. Both approaches are implemented to calculate the design sensitivities for continuum and structural isoparametric elements. To demonstrate the accuracy and robustness of the developed analytical approach compared to the semi-analytical approach some test cases using different structural and continuum element types are presented.

1 Introduction

A popular method for calculating sensitivity derivatives is the semi-analytical method which is based on the finite difference evaluation of the derivatives of the finite element stiffness matrix. Using either a forward or backward difference method to calculate the sensitivities of a structure for m design variables requires the evaluation of $m+1$ stiffness matrices. In addition to the time consideration, these methods are also sensitive to changes in step size.

One alternative approach that is generating much interest is to calculate the derivatives analytically using the implicit differentiation techniques from which the semi-analytical approach is obtained. The issue of which approach of obtaining sensitivity data is best is a much debated subject. References [1-5] present comparisons of the various methods and their relative merits. The early work done by Zienkiewicz and Campbell [6] and Ramakrishnan and Francavilla [7] has been refined and extended over the last one and a half decades. More recent work by Wang, Sun, and Gallagher [8] in 1985 has provided formulations employing an implicit differentiation approach for sensitivity analysis of 2-D and 3-D isoparametric continuum elements. Other work by Brockman and Lung [9], 1988, presents an approach for the sensitivity analysis of Mindlin plate and shell elements. The formulations presented by-pass the direct calculation of the stiffness matrix derivatives and thus yields a significant improvement in efficiency.

The purpose of this paper is to present an efficient approach to obtain the sensitivity of isoparametric continuum and structural elements to geometric properties and to compare it with the semi-analytical approach. The analytical formulation is presented in a general manner and is applicable to any isoparametric element type. Particular attention has been paid to the reduction of the number of calculations required and to the ease of implementation of the method into a general purpose finite element code.

2 Formulation of Displacement Derivatives

Consider the general formulation of a linear static finite element problem whose equations are of the form

$$K U = F \tag{1}$$

where K is the reduced global stiffness matrix for the structure, U is the vector of nodal displacements to be computed, and F is the vector of applied nodal forces.

The derivative of equation (1) with respect to any design variable a_m is given by

$$\frac{\partial K}{\partial a_m} U + K \frac{\partial U}{\partial a_m} = \frac{\partial F}{\partial a_m} \tag{2}$$

rearranging yields

$$K \frac{\partial U}{\partial a_m} = \frac{\partial F}{\partial a_m} - \frac{\partial K}{\partial a_m} U \tag{3}$$

The unknown terms of equation (3) are $\frac{\partial K}{\partial a_m}$ and $\frac{\partial F}{\partial a_m}$. The derivative, $\frac{\partial F}{\partial a_m}$, represents the sensitivity of the applied forces to the design variables. In many cases the applied forces are independent of the design variables and this term is zero. This assumption will be made for the remainder of the development. (A treatment for the formulation of $\frac{\partial F}{\partial a_m}$ in non-zero cases such as the sensitivity of pressure loading to shape variables is given in detail by Wang, Sun, and Gallagher in [8].) Thus equation (3) becomes

$$K \frac{\partial U}{\partial a_m} = - \frac{\partial K}{\partial a_m} U \tag{4}$$

Notice that the left-hand side of equation (4) has the same coefficient matrix as equation (1). Since the decomposed form of this matrix is normally stored for the calculation of multiple load cases, it may be reused to obtain $\frac{\partial U}{\partial a_m}$ in an analogous manner from the equation

$$\frac{\partial U}{\partial a_m} = K^{-1} \left\{ - \frac{\partial K}{\partial a_m} U \right\} \tag{5}$$

i.e. treating the vector $\left\{ - \frac{\partial K}{\partial a_m} U \right\}$ as a pseudo-load.

2.1 Displacement Derivative for Continuum Elements

Consider the formulation of the 2-D and 3-D isoparametric continuum elements. The stiffness matrix of the element is given by

$$K^e = \int_\Omega B^T C B |J| d\Omega \tag{6}$$

where Ω is the domain of the element, B is the element strain-displacement matrix, C is the material properties matrix, and $|J|$ is the determinate of the Jacobian matrix, J, which represents the mapping of the global coordinates X,Y, and Z into the element natural coordinates r, s, and t. For the continuum element, C is a function of the material properties only, i.e. Young's modulus and Poisson's ratio. Therefore the derivatives of C with respect to the shape and sizing parameters are zero and

$$\frac{\partial K^e}{\partial a_m} = \int_\Omega \left[\frac{\partial B^T}{\partial a_m} C B + B^T C \frac{\partial B}{\partial a_m} \right] |J| d\Omega + \int_\Omega \left[B^T C B \frac{\partial |J|}{\partial a_m} \right] d\Omega \tag{7}$$

Multiplying both sides of equation (7) by the local displacement vector, u, as in [7,9], results in the following

$$\frac{\partial K^e}{\partial a_m} u = \int_\Omega \left\{ \frac{\partial B}{\partial a_m} C B u + B^T C \frac{\partial B}{\partial a_m} u \right\} |J| d\Omega + \int_\Omega B^T C B u \frac{\partial |J|}{\partial a_m} d\Omega \tag{8}$$

but the stress vector, σ, is given by

$$\sigma = C B u \tag{9}$$

substituting (9) into (8) and rearranging

$$\frac{\partial K^e}{\partial a_m} u = \int_\Omega B^T \left\{ \sigma \frac{\partial |J|}{\partial a_m} + \sigma^* |J| \right\} d\Omega + \int_\Omega \frac{\partial B^T}{\partial a_m} \sigma |J| d\Omega \tag{10}$$

where

$$\sigma^* = C \frac{\partial B}{\partial a_m} u \tag{11}$$

Finally, by factoring $|J|$

$$\frac{\partial K^e}{\partial a_m} u = \int_\Omega \left[B^T \left\{ \sigma \frac{|J|'}{|J|} + \sigma^* \right\} + \frac{\partial B^T}{\partial a_m} \sigma \right] |J| d\Omega \tag{12}$$

where

$$|J|' = \frac{\partial |J|}{\partial a_m}$$

In order to evaluate equation (12) only two new quantities need be obtained, $\frac{\partial B}{\partial a_m}$ and $\frac{|J|'}{|J|}$. As will be shown later, significant savings are gained in the direct calculation of the quantity $\frac{|J|'}{|J|}$. Since these derivatives are also required for the structural elements, formulations for these terms will be given following their development. The remaining values will have

594

been evaluated in the generation of the stiffness matrix. Also, the form of equation (10) is the same as the expression for the stress vector. This will allow σ^* to be evaluated by re-using the same stress recovery routines.

2.2 Displacement Derivative for Structural Elements

The displacement derivatives for the shell and beam elements are derived in the same manner. The element stiffness matrix for an isoparametric beam or shell following the formulation in Bathe [10] has the form

$$K^e = \int_\Omega B^T Q^T C_a Q B \, |J| \, d\Omega \tag{13}$$

where Q is the matrix which transforms the element aligned material properties matrix C_a into the global coordinate system. Taking the derivative with respect to a_m

$$
\begin{aligned}
\frac{\partial K^e}{\partial a_m} &= \int_\Omega \left(\frac{\partial B}{\partial a_m}^T Q^T C_a Q B + B^T Q^T C_a Q \frac{\partial B}{\partial a_m} \right) |J| \, d\Omega \\
&+ \int_\Omega \left(B^T \frac{\partial Q}{\partial a_m}^T C_a Q B + B^T Q^T C_a \frac{\partial Q}{\partial a_m} B \right) |J| \, d\Omega \\
&+ \int_\Omega B^T Q^T C_a Q B \frac{\partial |J|}{\partial a_m} \, d\Omega
\end{aligned}
\tag{14}
$$

Using the same procedure as before we post multiply by u and simplify to obtain

$$\frac{\partial K^e}{\partial a_m} u = \int_\Omega \left\{ B^T \left[\sigma \frac{|J|'}{|J|} + \sigma^* + \tilde{\sigma} \right] + \frac{\partial B}{\partial a_m}^T \sigma \right\} |J| \, d\Omega \tag{15}$$

where

$$C = Q^T C_a Q \tag{16}$$

and

$$\tilde{\sigma} = 2 \hat{C}_S B u$$

$$\hat{C}_S = \left[Q^T C \frac{\partial Q}{\partial a_m} \right]_S \tag{17}$$

2.3 Formulation of Component Derivatives

Recall from the general finite element formulation that the strain-displacement matrix, B, is constructed by rearranging and adding the derivatives of the element interpolation functions N_K with respect to the global coordinates X, Y, and Z in accordance with the kinematic equations of the element formulation. For example, the three dimensional elasticity strain-displacement matrix is defined as

$$B = [\, b_1 \quad b_2 \quad \ldots \quad b_n \,] \tag{18}$$

$$b_k = \begin{bmatrix} N_{k,X} & 0 & 0 \\ 0 & N_{k,Y} & 0 \\ 0 & 0 & N_{k,Z} \\ N_{k,Y} & N_{k,X} & 0 \\ 0 & N_{k,Z} & N_{k,Y} \\ N_{k,Z} & 0 & N_{k,X} \end{bmatrix} \tag{19}$$

where n is the number of element nodes.

With the isoparametric formulation, the interpolation functions N are expressed in terms of the element natural coordinates $r, s,$ and t. Thus, the derivatives with respect to the global coordinates are evaluated by

$$\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} = J^{-1} \begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix} \tag{20}$$

where J^{-1} is the inverse Jacobian. Taking the derivative with respect to the design variable a_m yields

$$\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} = \frac{\partial J^{-1}}{\partial a_m}\begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix} + J^{-1}\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix} \qquad (21)$$

Since J^{-1} is not obtained in explicit form, an expression for $\frac{\partial J^{-1}}{\partial a_m}$ cannot be obtained directly. By using the identity

$$\frac{\partial J^{-1}}{\partial a_m}J + J^{-1}\frac{\partial J}{\partial a_m} = 0 \qquad (22)$$

the expression for $\frac{\partial J^{-1}}{\partial a_m}$ is found to be

$$\frac{\partial J^{-1}}{\partial a_m} = -J^{-1}\frac{\partial J}{\partial a_m}J^{-1} \qquad (23)$$

Thus equation (21) becomes

$$\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} = -J^{-1}\frac{\partial J}{\partial a_m}J^{-1}\begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix} + J^{-1}\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix} \qquad (24)$$

Substituting equation (20) into (24)

$$\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} = -J^{-1}\frac{\partial J}{\partial a_m}\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} + J^{-1}\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix} \qquad (25)$$

Equation (25) contains two derivatives that must be evaluated. First consider $\frac{\partial J}{\partial a_m}$. The Jacobian is obtained from the matrix multiplication

$$J = \begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix}\{X_k \quad Y_k \quad Z_k\} \qquad (26)$$

Applying the chain rule, the derivative is

$$\frac{\partial J}{\partial a_m} = \begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix}\frac{\partial}{\partial a_m}\{X_k \quad Y_k \quad Z_k\} + \frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix}\{X_k \quad Y_k \quad Z_k\} \qquad (27)$$

Substituting equation (27) into (25)

$$\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} = J^{-1}\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix} - \left[\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix}\frac{\partial}{\partial a_m}\{X_k \quad Y_k \quad Z_k\} + J^{-1}\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,r} \\ N_{k,s} \\ N_{k,t} \end{Bmatrix}\{X_k \quad Y_k \quad Z_k\}\right]\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} \qquad (28)$$

In equation (28), if the element interpolation functions are dependant on the geometric parameters as is the case for the structural elements presented, then all terms must be evaluated. However, if the element interpolation functions are not dependant on the geometric properties, then equation (28) simplifies to a form equivalent to that presented in [9].

$$\frac{\partial}{\partial a_m}\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} = -\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix}\frac{\partial}{\partial a_m}\{X_k \quad Y_k \quad Z_k\}\begin{Bmatrix} N_{k,X} \\ N_{k,Y} \\ N_{k,Z} \end{Bmatrix} \qquad (29)$$

596

The scalar, $\frac{|J|'}{|J|}$ can be generated directly without evaluating $\frac{\partial |J|}{\partial a_m}$ explicitly. The derivative of the determinate of the Jacobian may be written as

$$\frac{\partial |J|}{\partial a_m} = \frac{\partial |J|}{\partial J_{jk}} \frac{\partial J_{jk}}{\partial a_m}$$

(30)

where the repeated indices follow the conventional summation rule. By performing a term by term expansion of $\frac{\partial |J|}{\partial J_{jk}}$ it can be demonstrated that

$$\frac{\partial |J|}{\partial J_{jk}} = J_{kj}^{-1} |J|$$

(31)

Substituting (31) into equation (30)

$$\frac{\partial |J|}{\partial a_m} = J_{kj}^{-1} |J| \frac{\partial J_{jk}}{\partial a_m}$$

(32)

Rearranging yields

$$\frac{|J|'}{|J|} = J_{jk}^{-1} \frac{\partial J_{jk}}{\partial a_m} = \text{trace}\left(J^{-1} \frac{\partial J}{\partial a_m} \right)$$

(33)

This result is significant since this product is evaluated as a part of equations (28) or (29) and thus equation (33) is evaluated with only the two or three additions necessary to compute the trace.

3 Numerical Considerations for Analytical and Semi-Analytical Approaches
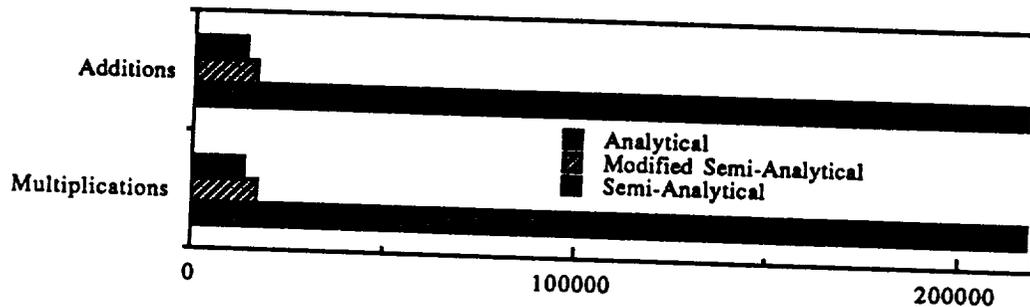
Figure 1—Comparison of Number of Operations Performed for 20-noded Solid Element.

From an implementation standpoint, the preceding approach for obtaining sensitivity derivatives is considerably more complex than the more conventional semi-analytical approach currently in wide use. As a result of this complexity, the analytical approach is more element dependant and requires the addition of more program statements to the basic finite element code. The increased efficiency and accuracy, however, more than offset the added programing effort.

Consider the calculation of the pseudo-load vector for a single 3-D solid element. The approach presented evaluates the pseudo-load vector with $(216n + 486)$ multiplications and $(216n + 414)$ additions, where n is the number of degrees of freedom in the element. Using the conventional semi-analytical method, where the stiffness matrix derivative in equation (5) is calculated directly by either a forward or backward difference method, assembled and then post-multiplied by the global displacement vector, requires $(55n^2 + 324n + 351)$ multiplications and $(55n^2 + 324n + 321)$ additions. This can be improved considerably by using equation (10) where the derivatives of the strain-displacement and Jacobian matrices are calculated via finite difference. This reduces the number of operations to $(270n + 811)$ multiplications and $(270n + 513)$ additions. This approach will be referred to as the modified semi-analytical approach to differentiate it from the more conventional approach. Figure 1 shows the total number of calculations involved for a 20-noded solid element for each of these approaches.

4 Implementation and Test Cases

To study the accuracy and efficiency of the analytical approach compared with the semi-analytical approach, the algorithms were implemented and several test problems analyzed. Test cases were chosen with several goals in mind. The main criteria was that the structures had easily verifiable sensitivity data. A second criteria for the test cases was that the structures could reasonably be modeled with several different types of elements.

4.1 Test Problem 1: Cantilever Beam

A simple cantilever beam is considered as an example to compare the accuracy of the two approaches for obtaining design sensitivities. The geometry, loading and material properties of the beam modeled are shown in figure 2. The 100 lb. load is applied parabolically to the right end of the beam. The design variable, a, is chosen to be the height of the beam. It is desired to calculate the sensitivity of the vertical displacement, v, of the beam at point A.
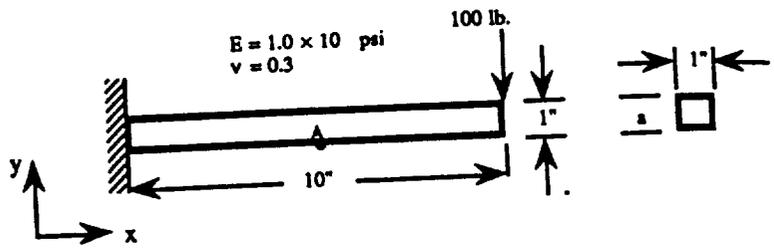
Figure 2—Cantilever Beam.

The cantilever beam is modeled by cubic beam, parabolic plane stress, parabolic shell and parabolic solid isoparametric elements. Figure 3 shows the displacement results graphically. Table 1 summarizes the sensitivity results obtained with the analytical approach for various elements and with the semi-analytical approach for the plane stress model. The last two columns of the table show the values obtained using different increments for the forward difference. This demonstrates the major drawback of using finite difference based derivatives — choosing the proper step size. If the step size is too large or too small, then errors will result. The range of acceptable step sizes varies form problem to problem, so that there will always be some doubt about the accuracy of these derivatives. In this example, semi-analytical approach converged to the value obtained by the analytical approach for the smaller step size, but there is considerable error with the larger step size. As can be seen in figure 4 the values for the sensitively of the vertical displacement at point A converge rapidly as the mesh density increases and are very close for all of the element formulations with the exception of the semi-analytical approach using $\Delta = .001$.
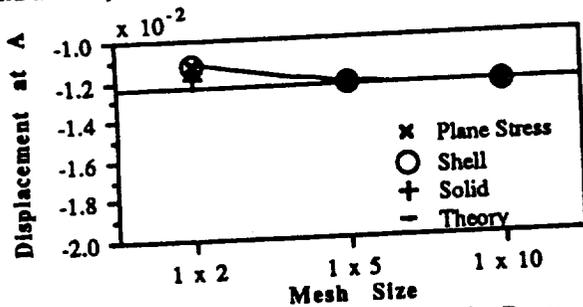
Figure 3—Accuracy of Displacement for Beam.
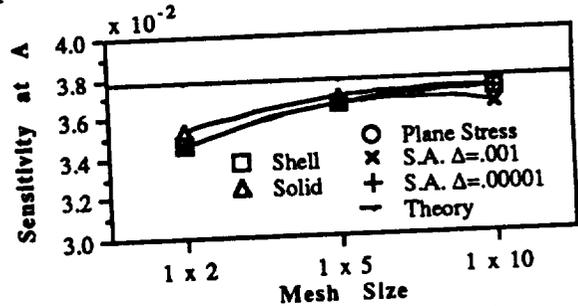
Figure 4—Accuracy of Sensitivity for Beam

				dv_A/db (10^{-2})
					Plane Stress
Mesh	Beam	Shell	Solid	Analytical	S.A. $\Delta=.001$	S.A. $\Delta=.00001$
1 x 2	3.766	3.456	3.531	3.456	3.453	3.456
1 x 5	——	3.654	3.685	3.654	3.649	3.653
1 x 10	——	3.716	3.723	3.716	3.640	3.716
theory				3.735		

Table 1—Accuracy of Displacement Sensitivity for Cantilever Beam.

4.2 Test Problem 2: Two Dimensional Plate

A simple two-dimensional plate is considered as an example to compare the relative efficiency of the analytical approach and the semi-analytical approach. The geometry of the plate, loading and material properties are shown in Figure 5. The external loading of 100 lb. is applied parabolically to the right side. The design variable, a, was chosen in this case to be the vertical dimension of the plate. The vertical displacement and sensitivity at point A are calculated using parabolic isoparametric plane stress, shell and solid elements.

Numerical results for the displacement and sensitivity at point A in the vertical direction for six mesh densities are given in Table 2. Columns 2 through 4 represent the displacement at A for the different finite element models used. Columns 5 through 8 show the sensitivity of the vertical displacement at point A calculated by the different finite element models and approaches. Figure 6 shows the displacement results. Figure 7 shows
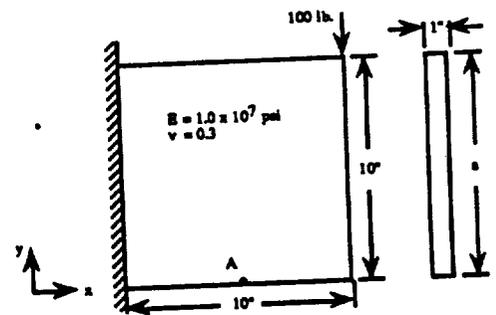
Figure 5—Plate.

the sensitivity results of Table 2. Again, the results obtained by the analytical approach converged well. Also, notice that in this case the step size Δ=.001 was sufficient to provide acceptable results.

The test cases were run on a Macintosh II personal computer. Figure 8 shows the execution times of each approach for a plane stress analysis of the plate. As expected the best performance was obtained in all cases using the analytical approach. The modified semi-analytical approach was a close second, but always slightly slower.

Mesh	v_A (10^{-5}) in.			dv_A/da (10^{-6})			
	Shell	Solid	Plane Stress	Shell	Solid	Plane Stress Analytical	S.A. Δ=.001
1 x 1	−2.496	−2.496	−2.408	4.669	4.433	4.669	4.845
2 x 2	−2.761	−2.761	−2.697	5.380	5.178	5.380	5.173
3 x 3	−2.825	−2.825	−2.773	5.487	5.339	5.487	5.381
4 x 4	−2.837	−2.837	−2.879	5.490	5.608	5.490	5.409
5 x 5	−2.849	−2.849	−2.816	5.494	5.400	5.494	5.436
6 x 6	−2.856	−2.856	−2.828	5.495	5.318	5.495	5.456

Table 2—Accuracy of Displacement Sensitivity for Thin Plate.
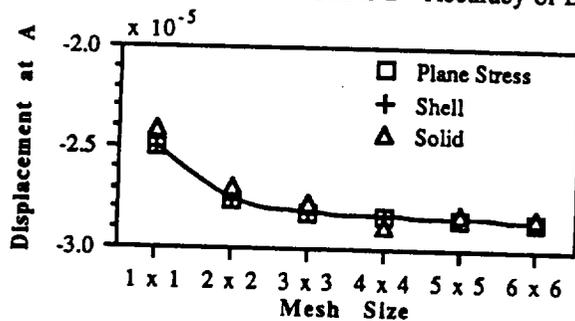
Figure 6—Accuracy of Displacement for Plate.
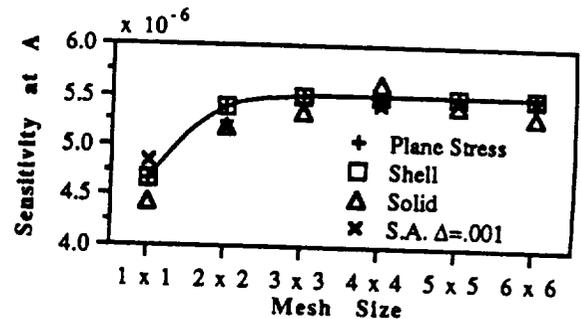
Figure 7—Accuracy of Sensitivity for Plate
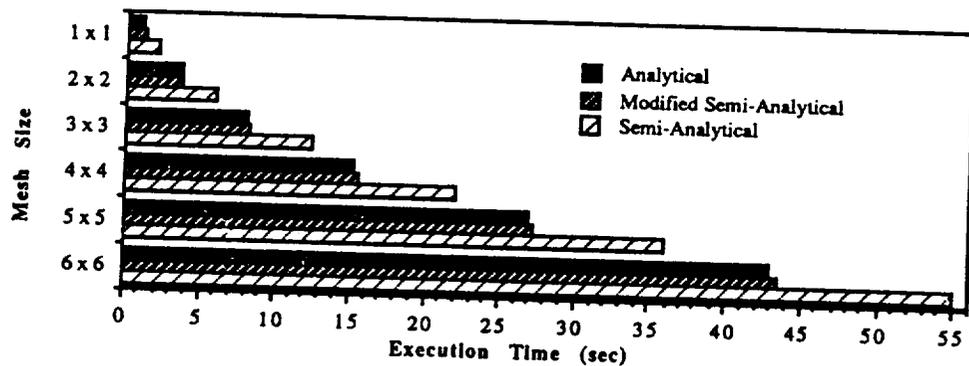
Figure 8—Execution Time for Plate Sensitivity Analysis

5 Conclusions

An analytical method was developed to incorporate design sensitivity derivatives into the finite element analysis. The sensitivity derivatives are obtained by implicit differentiation for the isoparametric element family of structural and continuum elements. The formulation allows for the calculation of any geometric sensitivity parameters of the element and can be implemented with relative ease since most of the required information is obtained in the course of the finite element solution.

In order to evaluate the merits of this approach, a comparison was made to the semi-analytical approach. The numerical test cases conducted show that the analytical approach is the more desirable approach for two reasons. It does not require the user to provide a proper step size as does the finite difference based semi-analytical approach and it is more efficient with respect to time.

REFERENCES

1. R. J. Yang and M. E. Botkin, 'The Relationship Between the Variational Approach and the Implicit Differentiation approach to Shape Design Sensitivities', *The Optimum Shape: Automated Structural Design*, ed. J. A. Bennett and M. E. Botkin, 1986, 61-77.

2. R. T. Haftka and H. M. Adelman, 'Recent Developments in Structural Sensitivity Analysis', *Structural Optimization*, Vol. 1, 1989, 137-151.

3. R. J. Yang and M. E. Botkin, 'Comparison Between the Variational and implicit differentiation Approaches to Shape Design Sensitivities', *AIAA Journal*, Vol. 24, 1986, 1027-1032.

4. R. J. Yang and M. E. Botkin, 'Accuracy of the Domain Method for the Material Derivative Approach to Shape Design Sensitivities', NASA Conference Publication 2457-*Sensitivity Analysis in Engineering*, 1987, 347-356.

5. R. Yang and K. K. Choi, 'Accuracy of Finite Element Based Shape Design Sensitivity Analysis', *Journal of Structural Mechanics*, Vol. 13, 1985, 223-239.

6. O. C. Zienkewicz and J. S. Campbell, 'Shape Optimization and Sequential Linear Programming', *Optimum Structural Design*, ed. R. H. Gallagher and O. C. Zienkiewicz, John Wiley & Sons, New York, 1973, 109-127.

7. C. V. Ramakrishnan and A. Francavilla, 'Structural Shape Optimization Using Penalty Functions', *Journal of Structural Mechanics*, Vol. 3, 1974, 403-422.

8. S. Y. Wang, Y. Sun and R. H. Gallagher, 'Sensitivity Analysis in Shape Optimization of Continuum Structures', *Computers & Structures*, Vol. 20, 1985, 855-867.

9. R. A. Brockman and F. Y. Lung, 'Sensitivity Analysis with Plate and Shell Finite Elements', *International Journal for Numerical Methods in Engineering*, Vol. 26, 1988, 1129-1143.

10. K. J. Bathe, *Finite Element Procedures in Engineering Analysis*, Prentice-Hall, Inc., 1982.

An Alternative Formulation

of the

Global Sensitivity Equations

Benjamin B. James

Lockheed Engineering & Sciences Company
144 Research Drive
Hampton, Virginia

Introduction

To optimize the performance of any system, the sensitivity derivatives of the system's output variables with respect to its input variables must be readily available. It is also desirable that these derivatives be inexpensive to calculate as the optimization process requires many evaluations of the output variables and their derivatives. Optimization methods that have been developed for use in automated structural design programs may not be extended for use in integrated multidisciplinary design programs until adequate means of calculating accurate sensitivity derivatives of complex, internally coupled systems have been developed. Until the development of the Global Sensitivity Equations (GSE) (Ref. 1), the only method of determining the sensitivity derivatives of coupled systems has been by using finite differences. Analytical or semi-analytical derivatives do not exist as there is no analytical solution to the coupled problem. Also, difficulties arise because the finite difference method is expensive as the system has to iterate to a converged solution for each incremental input variable. The method may not be accurate, and the choice of the input variable increment may cause the difference in the output variable to be insignificant compared to computer numerical error if the choice is too small, or the process may not predict the true value of the output variable if the increment is too large.

The GSE allow the system's sensitivity derivatives to be calculated as functions of the component subsystem's (local) sensitivity derivatives. These local sensitivity derivatives are calculated from specifically decoupled subsystems, whereas the GSE account for total system coupling. Since the subsystems are decoupled, it may be possible for the local derivatives to be calculated by analytical or semi-analytical methods, which generally reduce cost and improve accuracy. Several academic problems have been solved using GSE and have demonstrated encouraging results (Refs. 2, 3).

Approach

The formulation of the GSE (Ref. 1) is from a mathematical prospective. This paper will use an alternative formulation from an engineering prospective to develop the equations. This formulation will proceed in three steps: (1) Exact

problem definition; (2) Determination of required available information; and (3) Problem solution. Experience has demonstrated that once the problem is adequately defined and the known information is reduced to only that which is required, the solution is typically simplified. This simplification does exist for the development of the GSE.

GSE Formulation

Step 1: Problem Statement. In structural optimization approximate linear analyses using first order Taylor Series expansions to predict new behavior responses have been used successfully in various complex developments (Refs. 4, 5). It is assumed that this method can be extended for use in predicting the responses of a complex, internally coupled system in the region about a converged solution.

Figure 1 depicts the typical internal behavior of a system composed of three subsystems. The local inputs, $\vec{x}_i$, of a subsystem are subsets of and may be any, all, or none of the system inputs, $\vec{X}$. During the convergence process, the subsystems use the current values of the other subsystem responses, $\vec{y}_i$, as inputs. When the process has converged, the system output responses, $\vec{Y}$, are a union of subsets of the subsystem responses, $\vec{y}_i$.

New values of system behavior responses are predicted by

$$\vec{Y}(\vec{X}) = \vec{Y}_0(\vec{X}_0) + d\vec{Y}(\Delta\vec{X}) \tag{1}$$

The information required to predict the new values of the responses is the total differentials of the coupled system (Fig. 1),

$$dY_i = \sum_j \frac{\partial Y_i}{\partial X_j} dX_j \tag{2}$$

As the input differentials, dX_j, are chosen by some method, i.e., formal optimization, etc., the unknowns are the system derivatives, $\partial Y_i/\partial X_j$. The stated problem is to determine the system derivatives, $\partial Y_i/\partial X_j$.

The problem statement places two stringent assumptions on the solution. The first is that the system and its derivatives, and therefore, the subsystems, are linear. The second is that the solution process must start from a converged solution.

Step 2: Known Information. Figure 2 represents the decoupled system at a converged solution. The system may be decoupled since all of the inputs and outputs are known at the converged point. The total differentials of each subsystem, u_i, may be written as

$$d\vec{y}_1 = \frac{\partial u_1}{\partial \vec{x}_1} d\vec{x}_1 + \frac{\partial u_1}{\partial \vec{y}_2} d\vec{y}_2 + \frac{\partial u_1}{\partial \vec{y}_3} d\vec{y}_3$$

$$d\vec{y}_2 = \frac{\partial u_2}{\partial \vec{x}_2} d\vec{x}_2 + \frac{\partial u_2}{\partial \vec{y}_1} d\vec{y}_1 + \frac{\partial u_2}{\partial \vec{y}_3} d\vec{y}_3 \qquad (3)$$

$$d\vec{y}_3 = \frac{\partial u_3}{\partial \vec{x}_3} d\vec{x}_3 + \frac{\partial u_3}{\partial \vec{y}_1} d\vec{y}_1 + \frac{\partial u_3}{\partial \vec{y}_2} d\vec{y}_2$$

The differentials of the local input variables, $d\vec{y}_j$, and the local derivatives, $\partial u_i/\partial \vec{x}_j$ and $\partial u_i/\partial \vec{y}_k$, are known. The local input differentials correspond to the chosen system input differentials while the local derivatives are calculated. The local derivatives may be calculated by analytical, semi-analytical, or finite difference methods using the analysis capabilities of the decoupled subsystems. The unknowns are the total differentials of the output responses, $d\vec{y}_i$.

<u>Step 3: Problem Solution</u>. The set of equations (3) are the Global Sensitivity Equations. Rearranging and writing in convenient matrix notation gives the form presented in Reference 1, which is

$$\begin{bmatrix} I & \frac{\partial u_1}{\partial \vec{y}_2} & \frac{\partial u_1}{\partial \vec{y}_3} \\ \frac{\partial u_2}{\partial \vec{y}_1} & I & \frac{\partial u_2}{\partial \vec{y}_3} \\ \frac{\partial u_3}{\partial \vec{y}_1} & \frac{\partial u_3}{\partial \vec{y}_2} & I \end{bmatrix} \left\{ \begin{array}{c} d\vec{y}_1 \\ d\vec{y}_2 \\ d\vec{y}_3 \end{array} \right\} = \begin{bmatrix} \frac{\partial u_1}{\partial \vec{x}_1} & 0 & 0 \\ 0 & \frac{\partial u_2}{\partial \vec{x}_2} & 0 \\ 0 & 0 & \frac{\partial u_3}{\partial \vec{x}_3} \end{bmatrix} \left\{ \begin{array}{c} d\vec{x}_1 \\ d\vec{x}_2 \\ d\vec{x}_3 \end{array} \right\} \qquad (4)$$

As each entry in the local input vector, $\vec{x}_j$, corresponds to an entry in the system input vector, $\vec{X}$, equation (4) may be rewritten as

$$\begin{bmatrix} I & \frac{\partial u_1}{\partial \vec{y}_2} & \frac{\partial u_1}{\partial \vec{y}_3} \\ \frac{\partial u_2}{\partial \vec{y}_1} & I & \frac{\partial u_2}{\partial \vec{y}_3} \\ \frac{\partial u_3}{\partial \vec{y}_1} & \frac{\partial u_3}{\partial \vec{y}_2} & I \end{bmatrix} \left\{ \begin{array}{c} d\vec{y}_1 \\ d\vec{y}_2 \\ d\vec{y}_3 \end{array} \right\} = \begin{bmatrix} \frac{\partial u_1}{\partial X_1} & \frac{\partial u_1}{\partial X_2} & \cdots & \frac{\partial u_1}{\partial X_n} \\ \frac{\partial u_2}{\partial X_1} & \frac{\partial u_2}{\partial X_2} & \cdots & \frac{\partial u_2}{\partial X_n} \\ \frac{\partial u_3}{\partial X_1} & \frac{\partial u_3}{\partial X_2} & \cdots & \frac{\partial u_3}{\partial X_n} \end{bmatrix} \left\{ \begin{array}{c} dX_1 \\ dX_2 \\ \vdots \\ dX_n \end{array} \right\} \qquad (5)$$

By inverting equation (5), the coefficients for the system input differentials, $d\vec{X}$, are determined. The solution is completed by choosing the desired responses for $d\vec{Y}$ from the vectors of subsystem outputs, $d\vec{y}_i$.

Comments on the GSE

Generally, the right hand side matrix will not be fully populated as each system input variable will not be a local input variable for all subsystems.

The left hand side matrix is always a square matrix and invertible. Numerical problems may exist, however. Ill conditioning may exist if the values of the local derivatives differ by several orders of magnitude. Ill conditioning problems may be alleviated by scaling the derivatives to values close to unity before inverting and unscaling afterwards.

Examples

The full paper will contain several example problems. These problems will demonstrate several types of internal coupling and the resulting forms of the GSE.

REFERENCES

(1)   Sobieszczanski-Sobieski, J.: "On the Sensitivity of Complex, Internally Coupled Systems." AIAA paper no. 88-2378, April, 1988.

(2)   Sobieszczanski-Sobieski, J., Bloebaum, C., and Hajela, P.: "Sensitivity of Control-Augmented Structure Obtained by a System Decomposition Method." AIAA paper no. 88-2205, April 1988.

(3)   Hajela, P., Bloebaum, C., and Sobieszczanski-Sobieski, J.: "Application of Global Sensitivity Equations in Multidisciplinary Aircraft Synthesis." AIAA paper no. 89-2135, July 1989.

(4)   Schmidt, L. A. and Miura, H.: "Aproximation Concepts for Efficient Structural Synthesis." NASA CR-2552, March 1976.

(5)   Sobieszczanski-Sobieski, J., James, B. B., and Riley, M. F.: "Structural Sizing by Generalized, Multilevel Optimization." AIAA Journal, Vol. 25, No. 1, January 1987.
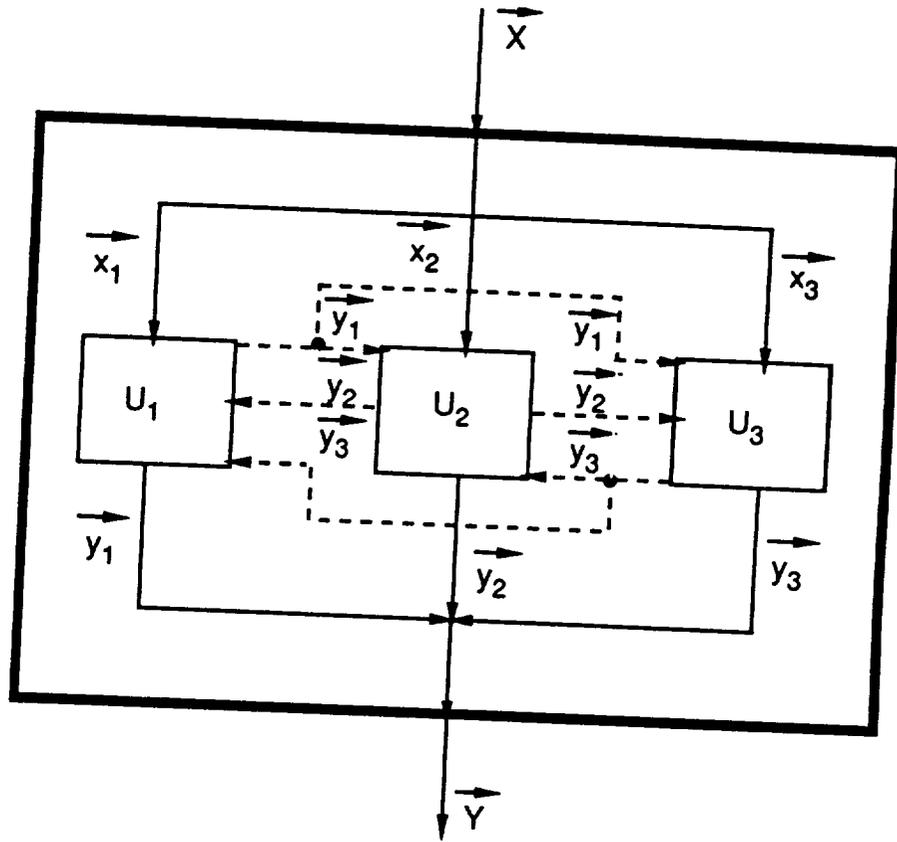
$$dY_i = \frac{\partial Y_i}{\partial X_1} \, dX_1 + \frac{\partial Y_i}{\partial X_2} \, dX_2 + \ldots + \frac{\partial Y_i}{\partial X_n} \, dX_n$$

Figure 1: System Total Differential

605
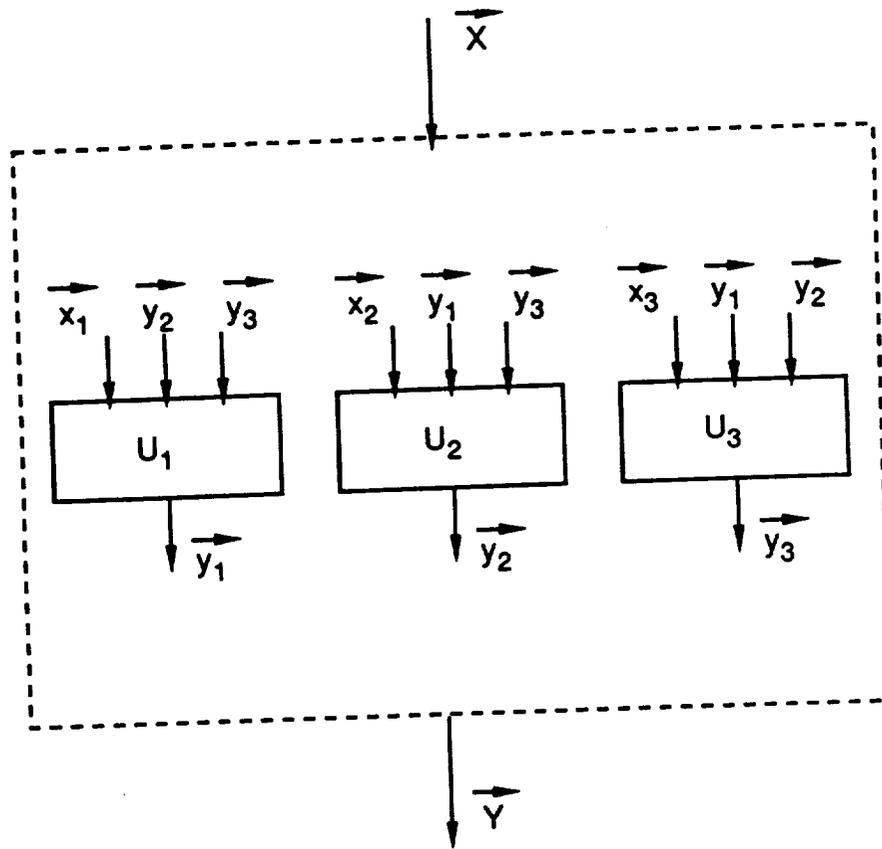
Decoupled Systems

$$\overrightarrow{dy_1} = \frac{\partial u_1}{\overrightarrow{\partial x_1}} \overrightarrow{dx_1} + \frac{\partial u_1}{\overrightarrow{\partial y_2}} \overrightarrow{dy_2} + \frac{\partial u_1}{\overrightarrow{\partial y_3}} \overrightarrow{dy_3}$$

$$\overrightarrow{dy_2} = \frac{\partial u_2}{\overrightarrow{\partial x_2}} \overrightarrow{dx_2} + \frac{\partial u_2}{\overrightarrow{\partial y_1}} \overrightarrow{dy_1} + \frac{\partial u_2}{\overrightarrow{\partial y_3}} \overrightarrow{dy_3}$$

$$\overrightarrow{dy_3} = \frac{\partial u_3}{\overrightarrow{\partial x_3}} \overrightarrow{dx_3} + \frac{\partial u_3}{\overrightarrow{\partial y_1}} \overrightarrow{dy_1} + \frac{\partial u_3}{\overrightarrow{\partial y_2}} \overrightarrow{dy_2}$$

Figure 2: Component Subsystem Total Differentials

THICKNESS AND ORIENTATIONAL DESIGN
FOR A MAXIMUM STIFF MEMBRANE

Pauli Pedersen

Department of Solid Mechanics

The Technical University of Denmark, Lyngby, Denmark

Abstract — Recent results from sensitivity analysis for strain energy with anisotropic elasticity are applied to thickness and orientational design of laminated membranes. Primarily the first order gradients of the total elastic energy are used in an optimality criteria based method. This traditional method is shown to give slow convergence with respect to design parameters, although the convergence of strain energy is very good. To get a deeper insight into this rather general characteristic, second order derivatives are included and it is shown how they can be obtained by first order sensitivity analysis. Examples of only thickness design, only orientational design and combined thickness—orientational design will be presented.

1. INTRODUCTION

Design with advanced materials, such as anisotropic laminates, is a challenging area for optimization. We shall here restrict ourselves to plane problems, as in the early work of BANICHUK [1] (which includes further early references). Recent work by the author [2],[3] was conducted independently and the formulations are rather parallel. Similar research is carried out by SACCHI LANDRIANI & ROVATI [4]. In the present paper we combine these orientational optimizations with thickness optimization. The further goal is to get a deeper insight into the redesign procedures based on optimality criteria.

The sensitivity analysis that proves local gradient determination relative to a fixed strain field is presented. The physical understanding of these results have many aspects outside the scope of the present paper. The early paper by MASUR [5] includes valuable information about this sensitivity analysis.

For orthotropic materials, a single optimization parameter controls the orientational design. This parameter includes information about material as well as about the state of strain. It is used as an optimization criterion and in principle, the optimization procedure is a non—gradient technique. In this way local extrema are avoided.

When the principal axes of an orthotropic material are equal to, say, the principal strain axes, it follows directly that principal stress axes also equal those of material and strain. However, optimal orientations exist for which the principal axes of material differ from those of the principal strains. Even for this case it is proved in [3] that the principal axes of stress equal those of the principal strains.

The sensitivity analysis for thickness change is extended to include the mutual sensitivities, i.e. change in energy density with respect thickness changes not at the same point. A symmetry relation is proven.

A number of actual examples will be shown and discussed, but are not included in this short pre—Conference paper.

2. SENSITIVITY ANALYSIS FOR ENERGY IN NON—LINEAR ELASTICITY

Let us start with the work equation

$$W + W^C = U + U^C \tag{2.1}$$

where W, W^C are physical and complimentary work of the external forces, and U, U^C are physical and complementary elastic energy, also named strain and stress energy, respectively.

The work equation (2.1) holds for any design h and therefore for the total differential quotient wrt. h

$$\frac{dW}{dh} + \frac{dW^C}{dh} = \frac{dU}{dh} + \frac{dU^C}{dh} \qquad (2.2)$$

Now in the same way as h represents the design field generally, ϵ represents the strain field and σ represents the stress field. Remembering that as a function of h, ϵ we have W, U, while the complementary quantities W^C, U^C are functions of h, σ. Then we get (2.2) more detailed by

$$\frac{\partial W}{\partial h} + \frac{\partial W}{\partial \epsilon}\frac{\partial \epsilon}{\partial h} + \frac{\partial W^C}{\partial h} + \frac{\partial W^C}{\partial \sigma}\frac{\partial \sigma}{\partial h} = \frac{\partial U}{\partial h} + \frac{\partial U}{\partial \epsilon}\frac{\partial \epsilon}{\partial h} + \frac{\partial U^C}{\partial h} + \frac{\partial U^C}{\partial \sigma}\frac{\partial \sigma}{\partial h} \qquad (2.3)$$

The principles of virtual work which hold for solids/structures in equilibrium are

$$\frac{\partial W}{\partial \epsilon} = \frac{\partial U}{\partial \epsilon} \qquad (2.4)$$

for the physical quantities with strain variation and for the complimentary quantities with stress variation we have

$$\frac{\partial W^C}{\partial \sigma} = \frac{\partial U^C}{\partial \sigma} \qquad (2.5)$$

Inserting (2.4) and (2.5) in (2.3) we get

$$\frac{\partial U^C}{\partial h} - \frac{\partial W^C}{\partial h} = -\left[\frac{\partial U}{\partial h} - \frac{\partial W}{\partial h}\right] \qquad (2.6)$$

and for design independent loads

$$\left[\frac{\partial U^C}{\partial h}\right]_{\substack{fixed \\ stresses}} = -\left[\frac{\partial U}{\partial h}\right]_{\substack{fixed \\ strains}} \qquad (2.7)$$

as stated by MASUR [5]. Note that the only assumption behind this is the design independent loads $\partial W / \partial h = 0$, $\partial W^C / \partial h = 0$.

To get further into a physical interpretation of $(\partial U / \partial h)_{fixed\ strains}$ (and by (2.7) of $(\partial U^C / \partial h)_{fixed\ stresses}$) we need the relation between external work W and strain energy U. Let us assume that this relation is given by the constant c

$$W = cU \qquad (2.8)$$

For linear elasticity and dead loads we have $c = 2$ and in general we will have $c > 1$.

Parallel to the analysis from (2.1) to (2.3) we based on (2.8) get

$$\frac{\partial W}{\partial h} + \frac{\partial W}{\partial \epsilon}\frac{\partial \epsilon}{\partial h} = c\frac{\partial U}{\partial h} + c\frac{\partial U}{\partial \epsilon}\frac{\partial \epsilon}{\partial h} \qquad (2.9)$$

that for design independent loads $\partial W / \partial h = 0$ with virtual work (2.4) gives

$$\frac{\partial W}{\partial \epsilon}\frac{\partial \epsilon}{\partial h} = \frac{\partial U}{\partial \epsilon}\frac{\partial \epsilon}{\partial h} = \frac{c}{1-c}\frac{\partial U}{\partial h} \qquad (2.10)$$

and thereby

$$\frac{dU}{dh} = \frac{\partial U}{\partial h} + \frac{\partial U}{\partial \epsilon}\frac{\partial \epsilon}{\partial h} = \frac{1}{1-c}\left[\frac{\partial U}{\partial h}\right]_{\substack{fixed \\ strains}} \qquad (2.11)$$

Note, in this important result that with $c > 1$ we have different signs for dU/dh and $(\partial U / \partial h)_{fixed\ strains}$.

For the case of linear elasticity and dead loads we have with $c = 2$ and adding (2.7)

$$\frac{dU}{dh} = -\left[\frac{\partial U}{\partial h}\right]_{\substack{fixed \\ strains}} = \left[\frac{\partial U}{\partial h}\right]_{\substack{fixed \\ stresses}} \tag{2.12}$$

For the case of non—linear elasticity by

$$\sigma = E\epsilon^n \tag{2.13}$$

and still dead loads $(W^C = 0)$ we get $c = 1+n$ and thereby

$$\boxed{\frac{dU}{dh} = -\frac{1}{n}\left[\frac{\partial U}{\partial h}\right]_{\substack{fixed \\ strains}} = \frac{1}{n}\left[\frac{\partial U}{\partial h}\right]_{\substack{fixed \\ stresses}}} \tag{2.14}$$

3. OPTIMALITY CRITERIA

We want to minimize the elastic strain energy U

$$\text{Minimize} \left[U = \sum_{e=1}^{N} U_e\right] \tag{3.1}$$

which is obtained as the sum of the element energies U_e for $e = 1,2,...,N$. Two groups of design parameters are considered. The material orientations θ_e for $e = 1,2,...,N$ assumed constant in each element, and the element thicknesses t_e for $e = 1,2,...,N$, also constant in each element. The constraint of our optimization problem is a given volume $\bar{V}$, i.e., by summation over element volumes V_e for $e = 1,2,...,N$

$$V - \bar{V} = \sum_{e=1}^{N} V_e - \bar{V} = 0 \tag{3.2}$$

The gradients of volume are easily obtained for thicknesses

$$\frac{\partial V}{\partial t_e} = \frac{\partial V_e}{\partial t_e} = \frac{V_e}{t_e} \tag{3.3}$$

and volume do not depend on material orientation

$$\frac{\partial V}{\partial \theta_e} = 0 \tag{3.4}$$

The gradients of elastic strain energy is simplified by the results of section two and thereby localized

$$\frac{\partial U}{\partial h_e} = -\left[\frac{\partial U}{\partial h_e}\right]_{fixed\ strains} = -\left[\frac{\partial(u_e\ V_e)}{\partial h_e}\right]_{fixed\ strains} \tag{3.5}$$

valid for $h_e = \theta_e$ as well as for $h_e = t_e$. The strain energy density u_e is introduced by $U_e = u_e V_e = u_e a_e t_e$ with a_e for element area.

With fixed strains, the thickness has no influence on the strain energy density u_e and thus with (3.3) and (3.5), we directly get

609

$$\frac{\partial U}{\partial t_e} = -\frac{U_e}{t_e} = -\frac{u_e V_e}{t_e} \tag{3.6}$$

With respect to material orientation the gradient is more complicated, because even with fixed strains will the energy density u_e depend on θ_e. A rather simple formula is derived in [2], in terms of principal strains ϵ_I, ϵ_{II} $\left(|\epsilon_I| > |\epsilon_{II}|\right)$ — angle ψ from direction of ϵ_I to principal material direction — and material parameters C_2 and C_3

$$\frac{\partial U}{\partial \theta_e} = \left[V(\epsilon_I - \epsilon_{II})^2 \sin 2\psi \left[C_2 \frac{\epsilon_I + \epsilon_{II}}{\epsilon_I - \epsilon_{II}} + 4C_3 \cos 2\psi \right] \right]_e \tag{3.7}$$

With the gradients determined by (3.3), (3.4), (3.6) and (3.7) we can now formulate optimality criteria. For the thickness optimization the well-known criterion of proportional gradients gives $-u_e V_e/t_e \sim V_e/t_e$ which means constant energy density, equal to the mean strain energy density $\bar{u}$

$$\boxed{u_e = \bar{u} \text{ for all } e} \tag{3.8}$$

See also the early paper by MASUR [5] for this optimality criterion.

For the material orientation optimization we have an unconstrained problem, and thus from (3.7) the optimality criterion

$$\boxed{\sin 2\psi \left[C_2 \frac{\epsilon_I + \epsilon_{II}}{\epsilon_I - \epsilon_{II}} + 4C_3 \cos 2\psi \right]_e = 0 \text{ for all } e} \tag{3.9}$$

How is a thickness distribution that fulfill (3.8) obtained? We shall discuss a practical procedure, cf. ROZVANY [6], which is based on a number of approximations. Firstly, we neglect the mutual influences from element to element, i.e. each element is redesigned independently (but simultaneously)

$$(t_e)_{\text{next}} = t_e + (\Delta t_e) \tag{3.10}$$

Secondly, the optimal mean energy density $\bar{u}$ is taken as the present mean energy density $\tilde{u}$. Thirdly, the element energy U_e is assumed constant through the change Δt_e and then from (3.8) we get

$$\frac{U_e}{V_e(1 + \Delta t_e/t_e)} = \tilde{u} \quad , \text{ i.e.} \tag{3.11}$$

$$\Delta t_e = t_e(u_e - \tilde{u})/\tilde{u} \quad \text{ or } \quad (t_e)_{\text{next}} = t_e \, u_e/\tilde{u}$$

It is natural to ask, why the gradient of element energy is not taken into account

$$(U_e)_{\text{next}} = U_e + \frac{\partial U_e}{\partial t_e} \Delta t_e \tag{3.12}$$

but this is explained by the fact that although $\partial U/\partial t_e$ is known by (3.6) the gradient of the local energy (the element strain energy)

$$\frac{\partial U_e}{\partial t_e} = \left[\frac{\partial U_e}{\partial t_e} \right]_{\text{fixed strain}} + \left[\frac{\partial U_e}{\partial \epsilon} \right] \frac{\partial \epsilon}{\partial t_e} \tag{3.13}$$

is more difficult to determine. The two terms in (3.13) have different signs, and also the other neglected terms $\partial U_e/\partial t_i$ for $e \neq i$ may be of the same order. Although the procedure (3.11) mostly work rather satisfactory, we shall extend our analysis to the coupled problem.

4. MUTUAL SENSITIVITIES

The redesign procedure by (3.11) neglect the mutual sensitivities, i.e. the change in element energy due to change in the thickness of the other elements. These sensitivities can be calculated by classical sensitivity analysis. Assume the analysis is related to a finite element model

$$[S]\{D\} = \{A\} \qquad (4.1)$$

where $\{A\}$ are the given nodal actions, $\{D\}$ the resulting nodal displacements and $[S] = \sum_e [S_e]$ the system stiffness matrix accumulated over the element stiffness matrices $[S_e]$ for $e = 1,2,...,N$.

Let h_e be an element design parameter without influence on $\{A\}$, then we get

$$[S] \frac{\partial\{D\}}{\partial h_e} = - \frac{\partial[S]}{\partial h_e} \{D\} = \{P_e\} \qquad (4.2)$$

where the right—hand side $\{P_e\}$ is a pseudo load, equivalent to design change. Knowing $\partial\{D\}/\partial h_e$ it is straight forward to calculate $\partial U_i/\partial h_e$. Generally the computational efforts correspond to one additional load for each design parameter.

Then with all the gradients $\partial U_e/\partial t_i$ available we can formulate a procedure for simultaneously redesign of all element thicknesses

$$\{t\}_{next} = \{t\} + \{\Delta t\} \qquad (4.3)$$

that takes the mutual sensitivities into account. In agreement with the optimality criteria (3.8) we change towards equal energy density $\bar{u}$ in all elements. Formulated in terms of strain energy per area we want

$$u_e t_e + \sum_{i=1}^{N} \frac{\partial(u_e t_e)}{\partial t_i} \Delta t_i = \bar{u}(t_e + \Delta t_e) \qquad (4.4)$$

or in matrix notation

$$\{ut\} + \big[\nabla(ut)\big]\{\Delta t\} = \bar{u}\Big[\{t\} + \{\Delta t\}\Big] \qquad (4.5)$$

with solution

$$\{\Delta t\} = \Big[\big[\nabla(ut)\big] - \bar{u}[I]\Big]^{-1} \Big\{(\bar{u}{-}u)t\Big\} \qquad (4.6)$$

The gradient matrix $\big[\nabla(ut)\big]$ consists of the quantities $\partial(u_e t_e)/\partial t_i$.

Note that with the assumption of fixed strain field, the strain energy per area is unchanged, i.e. $\big[\nabla(ue)\big] = [0]$ and we get the simple redesign formula (3.11). This procedure can therefore be evaluated by comparing the numerical values in the gradient matrix $\big[\nabla(ut)\big]$ with $\bar{u}$, especially the off—diagonal values.

An alternative formulation would be Newton—Raphson iterations directly on energy densities

$$(u_e{-}\bar{u}) + \sum_{i=1}^{N} \frac{\partial(u_e{-}\bar{u})}{\partial t_i} \Delta t_i = 0 \qquad (4.7)$$

or in matrix notation

$$[\nabla u]\{\Delta t\} = \bar{u}\{1\} - \{u\} \qquad (4.8)$$

Here, the gradient matrix $[\nabla u]$ constitutes $\partial u_e/\partial t_i$. An interesting formulation is obtained, when we multiply every row e with area a_e and get

$$[\nabla(ua)]\{\Delta t\} = \{(\bar{u}-u)a\} \tag{4.9}$$

The present matrix is now symmetric, which to the knowledge of the author is not well—known. Remembering that $u_e a_e = U_e/t_e$ we prove this directly from (3.6)

$$\frac{\partial^2 U}{\partial t_e \partial t_i} = - \frac{\partial(U_e/t_e)}{\partial t_i}$$

$$\frac{\partial^2 U}{\partial t_i \partial t_e} = - \frac{\partial(U_i/t_i)}{\partial t_e} \tag{4.10}$$

Therefore, as $\partial^2 U/(\partial t_e \partial t_i) = \partial^2 U/(\partial t_i \partial t_e)$ we have

$$[\nabla(ua)]^T = [\nabla(ua)] \tag{4.11}$$

5. CONCLUSION

Optimization problems with a single active constraint (thickness design with given volume) or without constraints (orientational design) can be solved by simple iterative redesigns based on derived optimality criteria.

For the thickness design this redesign procedure is studied by deriving higher order sensitivities. Second order sensitivities of total strain energy are evaluated as first order sensitivities of local (element) specific strain energy.

For the orientational design a normal gradient technique will generally not work, because many local optima exist. Therefore, design changes in each redesign must be based on a criterion that identifies the orientation which gives global minimum of strain energy.

For optimal material orientation we get coinciding principal stresses and strain directions. This is used as a "test optimality criterion", and can also be utilized during iteration.

Optimization of thickness distribution for anisotropic materials (and even a class of non—linearity too) is no more complicated than with simple linear isotropic materials. The criterion of uniform energy density still holds.

REFERENCES

[1]   Banichuk, N.V.: Problems and Methods of Optimal Structural Design, Plenum Press, New York, 1983.

[2]   Pedersen, P.: On Optimal Orientation of Orthotropic Materials, Structural Optimization, Vol. 1, 101—106, 1989.

[3]   Pedersen, P.: Bounds on Elastic Energy in Solids of Orthotropic Materials, Structural Optimization, Vol. 2, 55—63, 1990.

[4]   Landriani, G.S. & Rovati, M.: Optimal Design for 2—D Structures made of Composite Materials, subm. to J. of Engineering Materials and Technology.

[5]   Masur, E.F.: Optimum Stiffness and Strength of Elastic Structures, J. of the Engineering Mechanics Div., ASCE, EM5, 621—649, 1970.

[6]   Rozvany, G.I.N.: Structural Design via Optimality Criteria, Kluwer, 1989. 463 p.

FINITE ELEMENT ANALYSIS AND OPTIMIZATION
OF COMPOSITE STRUCTURES

by

Jan Thomsen

Institute of Mechanical Engineering, The University of Aalborg
Pontoppidanstraede 101, DK-9220 Aalborg East, Denmark

Abstract: Linearly elastic fiber reinforced composite discs and laminates in plane stress with variable local orientation and concentration of one or two fiber fields embedded in the matrix material, are considered. The thickness and the domain of the discs or laminates are assumed to be given, together with prescribed boundary conditions and in-plane loading along the edge.

The problem under study consists in determining throughout the structural domain the optimum orientations and concentrations of the fiber fields in such a way as to maximize the integral stiffness of the composite disc or laminate under the given loading. Minimization of the integral stiffness can also be carried out. The optimization is performed subject to a prescribed bound on the total cost or weight of the composite that for given unit cost factors or specific weights determines the amounts of fiber and matrix materials in the structure. Examples are presented by the end of the paper.

1. Introduction

This paper gives a brief account of recent research reported by the first author in [1] on optimization of fiber orientation and concentration in composite discs and laminates. The research is inspired by the initial work in the field by Rasmussen [2] (reported in Danish, account in English available in Niordson and Olhoff [3]) and by important recent developments of Pedersen [4-8]. Problems concerning optimization of fiber orientation have earlier been considered by Banichuk [9], and we refer to Sacchi Landriani and Rovati [10] for other current research activities in the area.

The motivation for the work described in this paper is that fiber reinforced composite materials are ideal for structural applications, where high stiffness and strength are required at low weight. Aircraft and spacecraft are typical weight sensitive structures, in which composite materials are cost effective. To obtain the full advantage of the fiber reinforcement, fibers must be distributed and oriented optimally with respect to the actual strain field. Hence, transfer of fiber material from initially lowly stressed parts of the body in order to strengthen the parts and directions that are subjected to large internal forces is the general idea of optimization of composite structures.

Thus, relative to refs. [4-10], we in this paper both use fiber orientations and -concentrations as design variables. Based on the strain field determined by finite element analysis we construct an iterative two-level optimization procedure that consists of an optimality criterion approach as described by Pedersen [4,5,8], and a mathematical programming technique. Here,

- in the first level, the local fiber orientations corresponding to a global optimum are determined using an optimality criterion for these design variables, and

- in the second level, the local distribution of the amounts of fiber and matrix materials available within a bound on total cost or weight, are determined on the basis of analytically derived design sensitivities. In this level, the optimization is carried out by means of a dual mathematical programming technique as implemented in the optimizer CONLIN by Fleury and Braibant [11].

2. Objective function

The integral stiffness of the composite structure will be selected as the objective function for optimization, and we will be primarily interested in maximization. The structure of maximum integral stiffness will be defined as the structure that has minimum total elastic strain energy subject to a given loading.

We shall assume that our composite disc or laminate can be locally considered as a macroscopically homogeneous, orthotropic material. The strain energy density u will then be given by the following formula for an orthotropic laminate, see e.g. Jones [12],

$$u = \frac{1}{2}\{\epsilon\}^T [A]\{\epsilon\} = \frac{1}{2}A_{11}\epsilon_{11}^2 + \frac{1}{2}A_{22}\epsilon_{22}^2 + A_{12}\epsilon_{11}\epsilon_{22} + 2A_{66}\epsilon_{12}^2 \tag{1}$$

where $\{\epsilon\} = \{\epsilon_{11}; \epsilon_{22}; 2\epsilon_{12}\}$ is the strain vector, and [A] the stiffness matrix.

We now use well-known formulas to express the strain component in (1) by the principal strains, ϵ_I and ϵ_{II}, and the angle ψ from the direction corresponding to the numerically largest principal strain ϵ_I ($|\epsilon_I| \geq |\epsilon_{II}|$) to the direction associated with the largest stiffness A_{11} ($A_{11} \geq A_{22}$), see Fig. 1.
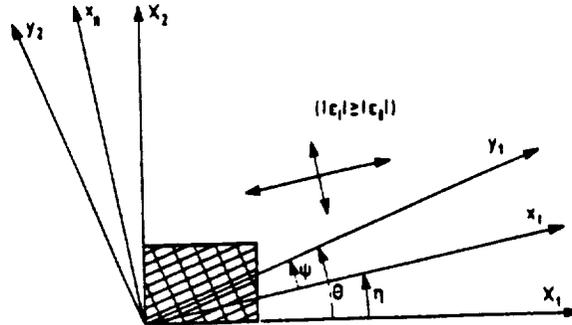
Fig. 1. Definition of the angles ψ, θ and η for mutual rotations of the finite element coordinate system X_1, X_2, the principal strain coordinate system x_I, x_{II} and the material coordinate system y_1, y_2

Since in the finite element analysis the structure is discretized into n elements with individual constant laminate stiffness matrices $[A]_i$, the total elastic strain energy U for the structure is then given by

$$U = \sum_{i=1}^{n} \left\{ \left\{ \frac{1}{8}A_{11}\left[(\epsilon_I + \epsilon_{II}) + (\epsilon_I - \epsilon_{II})\cos 2\psi\right]^2 + \frac{1}{8}A_{22}\left[(\epsilon_I + \epsilon_{II}) - (\epsilon_I - \epsilon_{II})\cos 2\psi\right]^2 \right. \right.$$

$$\left. \left. + \frac{1}{4}A_{12}\left[(\epsilon_I + \epsilon_{II})^2 - (\epsilon_I - \epsilon_{II})^2\cos^2 2\psi\right] + \frac{1}{2}A_{66}(\epsilon_I - \epsilon_{II})^2\sin^2 2\psi\right\} S \right\}_i$$

(2)

where S_i is the area of the i-th finite element.

3. Design model and cost function

The fiber orientation and concentration within each element of the discretized structure are adopted as design variables.

Our design model is made up of elements that consist of 3 fiber plies with the fiber orientations θ, $\theta + 90°$ and θ, and the volumetric fiber concentrations V_a, V_m and V_a, see Fig. 2.
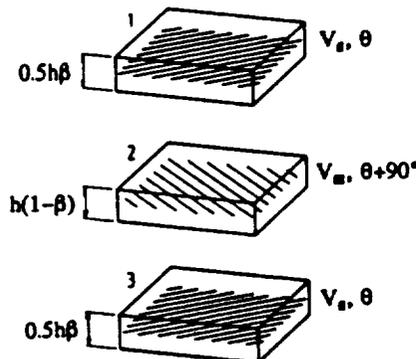
Fig. 2. Design variables of an element consisting of 3 orthogonal plies

Introducing the variable ratio β between the thickness of fiber ply 2 (in the middle) and the total thickness h of the element, we get the symmetric and orthotropic laminate shown in Fig. 2, which can have both unidirectional ($\beta=0$ v $\beta=1$) and cross ply ($0<\beta<1$) character.

We have now defined 4 design variables for each element: V_a, V_m, θ and β. For these design variables, we prescribe lower and upper constraint values as follows:

$$0 \leq (V_{fI})_i \leq \bar{V}_f \quad , \quad 0 \leq (V_{fII})_i \leq \bar{V}_f \quad , \quad 0 \leq \theta_i \leq 180° \quad , \quad 0 \leq \beta_i \leq 1$$

$$i = 1, \ldots, n \tag{3}$$

Here the given upper constraint value $\bar{V}_f$ for the fiber concentrations depends on how densely the fibers can be packed in the matrix material in view of their cross-sectional shape.

We finally formulate a constraint that enforces the total cost or weight C of the structure to be less than or equal to a given upper bound $\bar{R}$ if stiffness maximization is considered,

$$C = \sum_{i=1}^{n} \left\{ c_f \left[(V_{fI})_i h \beta_i + (V_{fII})_i h (1-\beta_i) \right] \right.$$

$$\left. + c_m \left[(1-(V_{fI})_i) h \beta_i + (1-(V_{fII})_i) h (1-\beta_i) \right] \right\} S_i \leq \bar{R} \tag{4}$$

Here c_f and c_m are given so-called "unit cost factors". They denote the cost per unit volume of the fiber and matrix materials, respectively, for a cost constrained problem, whereas c_f and c_m denote the specific weights of the fiber and matrix materials, respectively, if the total weight is constrained.

4. Stiffness matrix in terms of design variables

The fiber and matrix materials will be assumed to be linearly elastic with given Young's moduli E_f and E_m and Poisson's ratios v_f and v_m. We now adopt the "rule of mixtures", see e.g. Jones [12], for determining the components of the tensor of elasticity for a lamina in our design model

$$E_{Lj} = (1-V_{fj}) E_m + V_{fj} E_f \quad , \quad E_{Tj} = \frac{E_f E_m}{(1-V_{fj}) E_f + V_{fj} E_m}$$

$$\nu_{LTj} = (1-V_{fj}) \nu_m + V_{fj} \nu_f \quad , \quad G_{LTj} = \frac{G_m G_f}{(1-V_{fj}) G_f + V_{fj} G_m} \qquad j = I, II \tag{5}$$

Here indexes L and T refer to the longitudinal and transverse directions of the fibers, respectively, and in the following take on the "values" I and II that refer to the fiber layers 1 and 2, respectively.

For a composite element as shown in Fig. 2 that consists of 3 lamina with the thicknesses $0.5h\beta$, $h(1-\beta)$ and $0.5h\beta$ and the fiber orientations θ, $\theta+90°$ and θ, we can easily obtain the laminate stiffness matrix [A] by means of a formula given in Tsai & Pagano [13]. We get

$$[A] = \frac{E_{LI} h \beta}{8\alpha_{0I}} \left(\begin{bmatrix} 8 & 0 & 0 \\ s & 8 & 0 \\ & & 4 \end{bmatrix} + \alpha_{2I} \begin{bmatrix} \cos2\theta-1; & 0 & ;\sin2\theta/2 \\ s & -\cos2\theta-1; & \sin2\theta/2 \\ & & -1/2 \end{bmatrix} \right.$$

$$+ \alpha_{3I} \begin{bmatrix} \cos4\theta-1; & -\cos4\theta & ; & \sin4\theta \\ s & \cos4\theta-1 & ; & -\sin4\theta \\ & & -\cos4\theta-1/2 \end{bmatrix} + \alpha_{4I} \begin{bmatrix} 0 & 1 & 0 \\ s & 0 & 0 \\ & & -1/2 \end{bmatrix} \right)$$

$$+ \frac{E_{LII} h(1-\beta)}{8\alpha_{0II}} \left(\begin{bmatrix} 8 & 0 & 0 \\ s & 8 & 0 \\ & & 4 \end{bmatrix} + \alpha_{2II} \begin{bmatrix} \cos2\phi-1; & 0 & ;\sin2\phi/2 \\ s & -\cos2\phi-1; & \sin2\phi/2 \\ & & -1/2 \end{bmatrix} \right.$$

$$\left. + \alpha_{3II} \begin{bmatrix} \cos4\phi-1; & -\cos4\phi & ; & \sin4\phi \\ s & \cos4\phi-1 & ; & -\sin4\phi \\ & & -\cos4\phi-1/2 \end{bmatrix} + \alpha_{4II} \begin{bmatrix} 0 & 1 & 0 \\ s & 0 & 0 \\ & & -1/2 \end{bmatrix} \right) \tag{6}$$

where θ denotes the angle defining the fiber orientation, see Fig. 1, and the angle $\phi=\theta+90°$ defines the orthogonal direction.

The parameters $\alpha_0, \alpha_2, \ldots, \alpha_4$ in (6) can all be expressed explicitly in terms of the fiber concentrations V_{fj}, $j = I, II$, and the given elastic constants of the fiber and matrix materials, i.e.,

$$\alpha_{mj} = \alpha_{mj}(V_{fI}, V_{fII}, E_f, E_m, \nu_f, \nu_m) \qquad m = 0, 2, 3, 4 \qquad j = I, II \tag{7}$$

615

For reasons of brevity, the reader is referred to [1] for the specific expressions.

5. Optimization technique

The optimization problem is solved iteratively via a two-level procedure of redesign. The stress/strain field is initially determined by finite element analysis using MODULEF [14] in each loop of redesign, and improved orientations of the fibers are subsequently determined by means of an optimality criterion in the first level of redesign. In the second level of redesign the distributions of fibers are improved via a method of sensitivity analysis and mathematical programming.

A notable feature of the present problem is that a usual gradient method may fail in determining the optimal orientation of the fibers, because local optima normally exist, see e.g. Fig. 5.2 in [8]. To circumvent this inherent difficulty in the first level of redesign we follow Pedersen [4,8] and perform an analytical investigation of the first and second derivative in order to determine the global optimum of the total strain energy with respect to fiber orientation. From (2) and (6) we get the following expression for first order sensitivities, cf. Pedersen [4,8],

$$\frac{dU}{d\theta_i} - \frac{dU}{d\psi_i} = \left[4A\alpha_3 (\epsilon_I - \epsilon_{II})^2 \sin 2\psi (\gamma + \cos 2\psi) S \right]_i \quad , \quad i=1,\ldots,n \tag{8}$$

where A is a constant, and the parameter γ_i is defined by

$$\gamma_i = \left(\frac{\alpha_2}{4\alpha_3} \frac{1+\epsilon_{II}/\epsilon_I}{1-\epsilon_{II}/\epsilon_I} \right)_i \quad , \quad i=1,\ldots,n \tag{9}$$

The material parameters α_2 and α_3 are those appearing in (6). The results of a complete investigation of the extrema of U with respect to the key parameters ψ, α, and γ are summarized in a table in refs. [1],[4] and [8].

As described in [1],[4] and [8], the fiber orientation θ_i for each element can be determined by means of this table and the formula

$$\theta_i = \psi_i + \eta_i \quad , \quad i=1,\ldots,n \tag{10}$$

where η_i is the angle of rotation of the principal strain or stress direction of the i-th element relative to the X_1 axis of the finite element coordinate system, see Fig. 1.

The second stage in the loop of redesign consists in determining an improved distribution of the amount of fiber material, i.e., to obtain improved values of the design variables β_i, $(V_a)_i$, and $(V_m)_i$, (i=1,...,n). This is done by a dual method of mathematical programming using mixed variables as developed by Fleury and Braibant [14] and implemented in the computer code CONLIN. To this end we need the sensitivities of the objective function and constraints with respect to the aforementioned design variables.

Now, it is shown by Pedersen in [4,8] that by means of Clayperon's theorem and the principle of virtual displacements for structures with design independent loads, the gradient of the total strain energy can be determined from the gradient of the strain energy density u_i for a given element, whose strain field is considered to be fixed,

$$\frac{dU}{da_i} = - \frac{\partial u_i}{\partial a_i} S_i \quad , \quad i=1,\ldots,n \tag{11}$$

Here a_i denotes any of the design variables β_i, $(V_a)_i$, or $(V_m)_i$, i=1,...,n.

The sensitivities of the total strain energy U with respect to β_i, $(V_a)_i$ and $(V_m)_i$ can thus be determined by (2) and (11), assuming the strain field to be fixed, and restricting variation to the laminate stiffness matrix [A]. For the i-th element of the discretized geometry we then obtain the following expression for sensitivities w.r.t. the design variables a_i

$$U_{,a_i} = -\left\{ \left\{ \frac{1}{8}A'_{11} \left[(\epsilon_I + \epsilon_{II}) + (\epsilon_I - \epsilon_{II})\cos 2\psi \right]^2 + \frac{1}{8}A'_{22} \left[(\epsilon_I + \epsilon_{II}) - (\epsilon_I - \epsilon_{II})\cos 2\psi \right]^2 \right. \right.$$
$$\left. \left. + \frac{1}{4}A'_{12} \left[(\epsilon_I + \epsilon_{II})^2 - (\epsilon_I - \epsilon_{II})^2 \cos^2 2\psi \right] + \frac{1}{2}A'_{66}(\epsilon_I - \epsilon_{II})^2 \sin^2 2\psi \right\} S \right\}_i \qquad i=1,\ldots,n \tag{12}$$

where A'_{ij} is a shorthand notation for the derivatives dA_{ij}/da_i of a component A_{ij} of the stiffness matrix [A]. These sensitivities are derived analytically in [1], and the results are available therein. The sensitivities of the cost function (4) are readily derived analytically, and we thus have all the necessary sensitivity information that is required for the optimization in the second level of redesign.

6. Examples

We now consider two example problems of optimization of the rectangular composite disc shown in Fig. 3. The disc has one of its sides fixed against displacements in the X and Y directions, while the opposite side is subjected to a parabolically distributed shear loading.
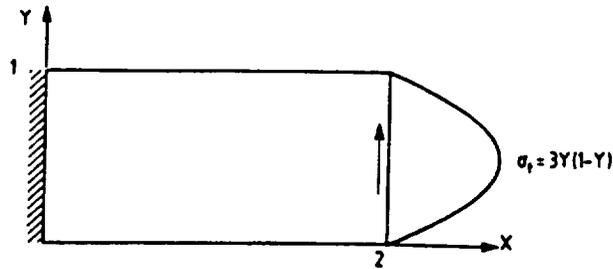
Fig. 3. Example problem for optimization

The upper constraint value $\bar{V}_f$ for fiber concentration in (3) is taken to be $\bar{V}_f = 80$ pct, and we only consider cases of $c_m = 0$ and $c_f = 1$, which means that the fibers are dominating in the cost or weight function C in (4).

In the first example we consider maximization of the stiffness of the disc under the condition that only one fiber field is allowed in each element. This corresponds to the special case of $\beta = 0$ v $\beta = 1$, see Chapter 3. The structure is discretized into $20 \cdot 40$ 4-node elements (type QUAD 2Q1D, see [14]). The result of the optimization is shown in Fig. 4, where the direction and density of the hatching within each elements illustrate the fiber orientation and concentration, respectively.
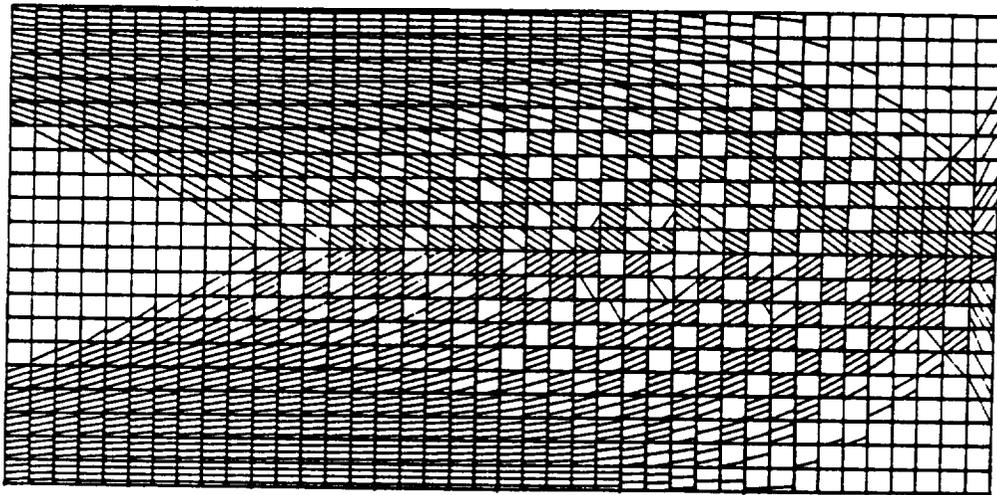
Fig. 4. Optimal distribution and orientation of fibers in first example: One fiber field, n=800, maximization of stiffness

We see that the lowly stressed elements do not contain any fibers. It is also noteworthy that the design contains "holes" in the fiber reinforcement in the mid part of the structure, where shear forces are dominating.

No doubt this is due to the fact that only one fiber field is allowed to exist in each element. This is not favourable in shear dominated areas with almost equal principal stresses, and the pattern obtained in the mid part may be conceived as the best possible attempt of the structure to increase its "shear force stiffness" under the given design conditions. The design shown in Fig. 4 is associated with a reduction of the total elastic energy U by 51% relative to the initial design, where all the fibers were uniformly distributed and given the orientation $\theta_i = 0$.

However, the convergence is very slow, and different designs may be obtained as a result of the optimization. In particular, the designs depend on the size of the applied FE-mesh, and it is not possible to obtain a limiting, numerically stable design by consecutively decreasing the mesh size. These features, along with the generation of "holes" in the design, indicate the necessity of a regularization of the formulation of the optimization problem (see, e.g., the survey by Olhoff and Taylor [15]).

This leads to our second example: Regularization of the formulation of the type of problem just considered is simply obtained by extending the design space such as to allow for formation of two orthogonal fiber fields everywhere in the disc (which is actually covered in the preceding chapters). Introducing two fiber fields, the design in Fig. 4 is replaced by the solution shown in Fig. 5, where the "shear force reinforcement" appears along the horizontal center line in agreement with

the boundary and symmetry conditions. Optimizing the structure, U is reduced by 55%. Now the convergence is rapid and the design is found to be independent of the discretization, which confirms that regularization has been achieved.
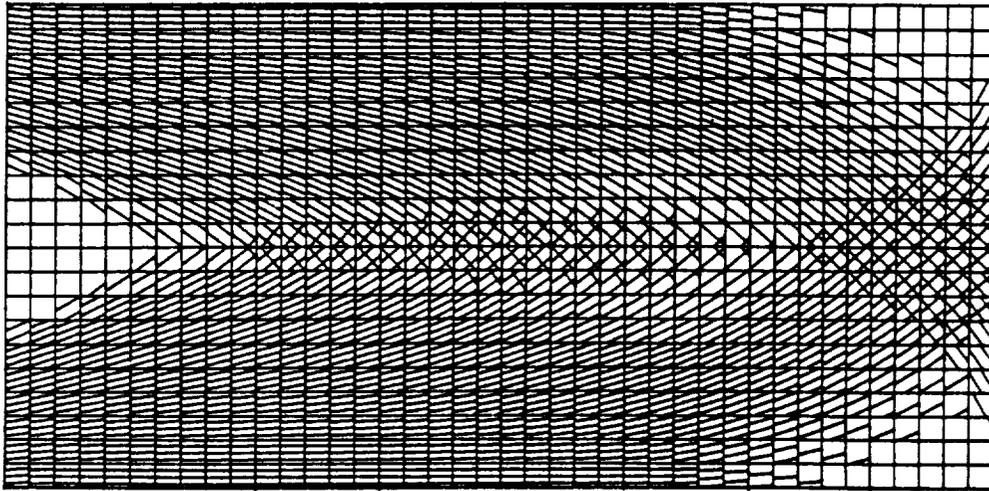
Fig. 5. Optimal distribution and orientation of fibers in second example: Two fiber fields, n=800, maximization of stiffness

References

[1] Thomsen, J.: Optimization of Composite Discs. Rept. no. 21, Institute of Mechanical Engineering, The University of Aalborg, Aalborg, Denmark, pp. 27, May 1990. (Submitted to Structural Optimization).

[2] Rasmussen, S.H.: Optimization of Fiber Reinforced Structures (in Danish), DCAMM report no. S12, The Technical University of Denmark, Lyngby, Denmark, pp. 127, 1979.

[3] Niordson, F.; Olhoff, N.: Variational Methods in Optimization of Structures. Trends in Solid Mechanics 1979, Proc. W.T. Koiter Anniversary Symp. (eds.: J.F. Besseling and A.M.A. van der Heijden), Sijthoff & Noordhoff, 177-194, 1979.

[4] Pedersen, P.: Bounds on Elastic Energy in Solids of Orthotropic Materials. Structural Optimization, 2, 55-63, 1990.

[5] Pedersen, P.: On Optimal Orientation of Orthotropic Materials. Structural Optimization. 1, 101-106, 1989.

[6] Pedersen, P.: Combining Material and Element Rotation in One Formula. Comm. Appl. Num. Meth. (to appear 1990).

[7] Pedersen, P.: On Laminate Layout for Rectangular Plates in Bending. The Tecnical University of Denmark, Lyngby Denmark.

[8] Pedersen, P.: On Material Orientation for Maximum Stiffness or Flexibility. The Tecnical University of Denmark, Lyngby Denmark.

[9] Banichuk, N.V.: Problems and Methods of Optimal Structural Design. New York: Plenum Press, 181-207, 1983.

[10] Sacchi Landriani, G.; Rovati, M.: Optimal Design of 2-D Structures Made of Composite Materials. (Submitted to J. Engrg. Materials and Technology).

[11] Fleury, C.; Braibant, V.: Structural Weight Optimization by Dual Methods of Convex Programming. International Journal for Numerical Methods in Engineering. 14, 1761-1783, 1986.

[12] Jones, R.M.: Mechanics of Composite Materials. New York: McGraw- Hill, 1975.

[13] Tsai, S.W.; Pagano, N.J.: Invariant Properties of Composite Materials. Composite materials workshop. Technomic, Westport Com, 1968.

[14] Institut National de Recherche en Informatique et en Automatique (INRIA, France). Avril 1983: Modulef, Bibliotheque D'Elasticite. 101, 1983.

[15] Olhoff, N.; Taylor, J.E.: On Structural Optimization, J. Appl. Mech., 1139-1151, 1983.

AUTHOR INDEX