N89-25176

RESEARCH ON OPTIMIZATION-BASED DESIGN AT THE ENGINEERING DESIGN METHODS LAB, BRIGHAM YOUNG UNIVERSITY

R.J. BALLING
A.R. PARKINSON
J.C. FREE
BRIGHAM YOUNG UNIVERSITY
PROVO, UTAH

ENGINEERING DESIGN METHODS RESEARCH LAB

To develop engineering design methods which

1) are domain-independent to a certain degree,

2) can be implemented on the computer.

THE OPTDES.BYU PACKAGE VERSION 4.0

A collection of general design methods that have been developed or collected by the Lab and implemented as software.

1) Use in the classroom -- about 80-100 students each year

2) Use in research -- about 12-15 graduate students each year

2) Use in industry -- about 90 company sites

OPTDES.BYU / COMMERCIAL ANALYSIS PACKAGES

1) OPTDES/ACSL (MGA Associates, Concord, MA)
   * simulation of dynamic systems

2) OPTDES/MECHANICAL ADVANTAGE (Cognition, Billerica, MA)
   * mechanical design software

3) OPTDES/COSMOSM (SRAC, Santa Monica, CA)
   * finite element analysis on microcomputers

4) OPTDES/CALIPER (Aptek, Colorado Springs, CO)
   * interference calculation and packing of geometric shapes

5) OPTDES/CIVILPAK (BYU)
   * design of land subdivisions, water distribution networks, steel frames, reinforced concrete systems

6) OPTDES/??? (Design Synthesis, Provo, UT)
   * SBIR Phase I Award from Wright-Patterson AFB to develop an optimization, feature-based modeling system for the design of mechanical parts

STANDARD CAPABILITIES OF OPTDES.BYU

1) First-class optimization algorithms:
   * Powell's Sequential Quadratic Programming Algorithm
   * Our Own Hybrid SQP/GRG Algorithm — an SQP that stays feasible
     - uses the SQP search direction (updated hessian)
     - uses the GRG line search (hemstitching)
   * SLP and Method of Centers
   * Goldfarb/Idnani's Dual Algorithm for QP problems
   * Revised Simplex Algorithm for LP problems
   * BFGS Variable Metric Algorithm for unconstrained problems

2) Interactive Design Utilities
   * Trial-And-Error Design (Set and Display)
   * History Backtrack and History Plots
   * 1D, 2D, and 3D Plots of Design Space or Subspace

3) Flexible Problem Setup
   * Function Designation (as objectives or constraints)
   * Many-to-One Variable and Function Mappings
   * Bounds on Variables and Allowable Values on Constraints
   * Log Variables and Functions

4) Interface with Analysis
   * Conventional and Generalized Interfaces
   * Programming-Free Interface

NEW AND DEVELOPING CAPABILITIES OF OPTDES.BYU

1) Manufacturing Considerations
   * Optimization with variables that are available in discrete combinations
   * Optimization in light of manufacturing tolerances on variables

2) Large-Scale Problems
   * Approximations based on analyses according to statistical test plans
   * Decomposition of optimization problems

3) Topological Optimization
   * Use of AI heuristic search strategies
   * Formal expert systems for generating topologies

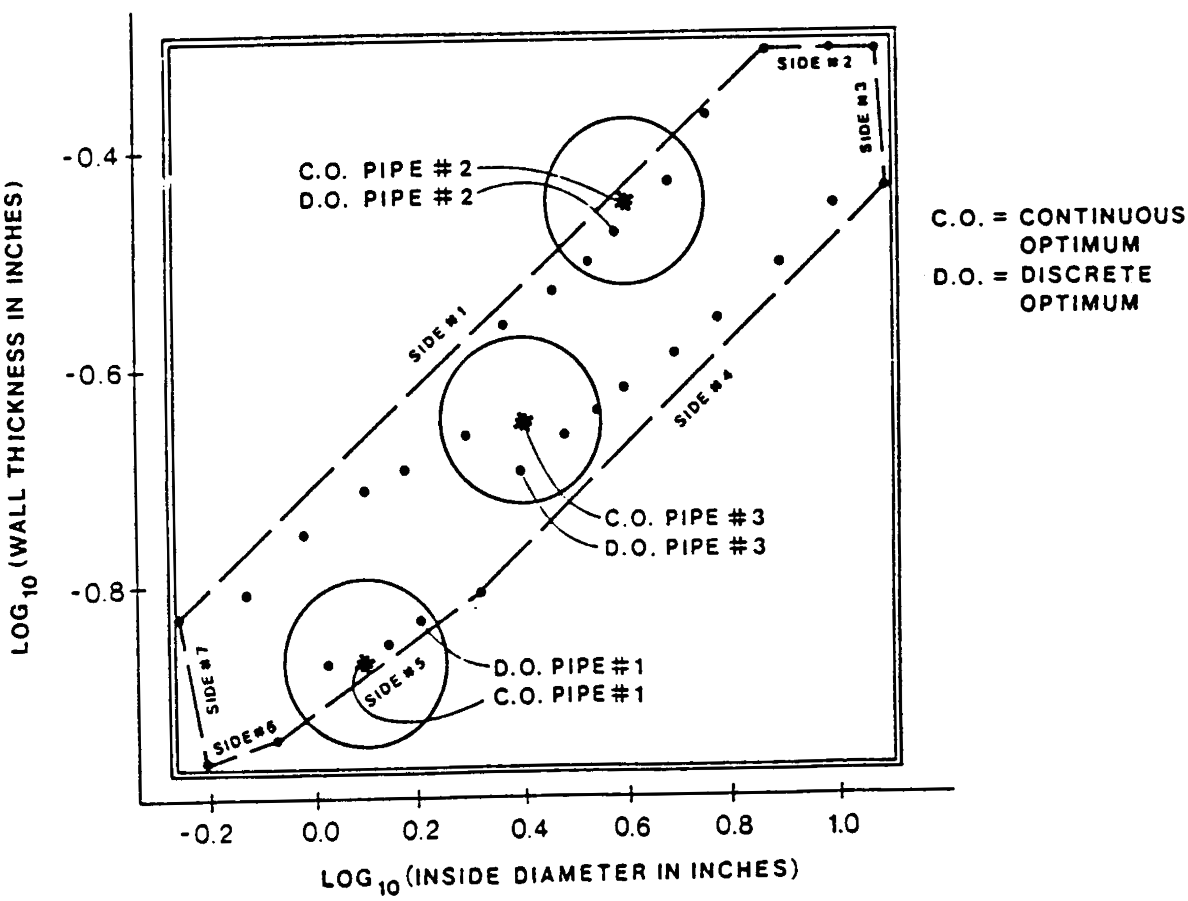
OPTIMIZATION WITH DISCRETE VARIABLES
STEP 1: CONTINUOUS OPTIMIZATION WITH ENVELOPING CONSTRAINTS

In this illustrative example it is desired to find the optimum diameters and thicknesses for three pipes. Each dot represents an available pipe. Log values are used to make the spacing of the dots more uniform.

A convex set of linear constraints are collapsed around the available pipes. This means 7x3=21 constraints are added to the optimization problem. Constraints may be thrown out by the user in the order of smallest length (or area in 3D, or hyper-area in ND). In this problem side #6 would be the first to be thrown out, followed by side #2, then side #3, and so on.

The continuous optimization problem is solved, and continuous optimum values are indicated by asterisks.

Neighborhoods are constructed about continuous optimum values. The user selects the radii to be large enough to include a sufficient number of dots, but small enough to keep the computational search effort reasonable.



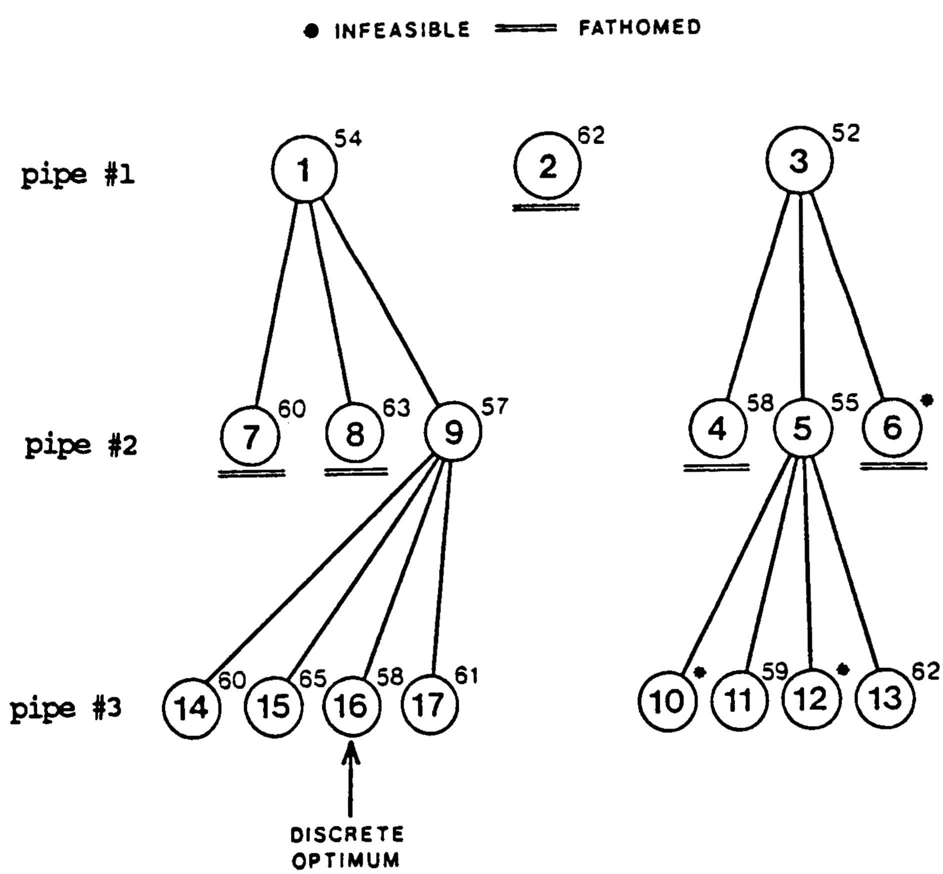
OPTIMIZATION WITH DISCRETE VARIABLES
STEP 2: DISCRETE SEARCH

In the illustrative example, the neighborhoods included 3 dots for pipe #1, 3 dots for pipe #2, and 4 dots for pipe #3. The number of possible discrete designs are 3x3x4=36. Exhaustive search would require 36 analyses. Not all these designs need be considered if a branch-and-bound strategy is used. The first step is to linearize the objective and constraints about the continuous optimum.

The figure shows a schematic of a branch-and-bound strategy. The circles represent "nodes" which are numbered in their order of generation. Nodes 1-3 represent designs where pipe #1 is fixed to the respective three dots within its neighborhood while pipes #2 and #3 remain continuous.

The small numbers beside each node represent the minimum costs for these designs. These values are found by optimizing continuous pipes #2 and #3 for minimum cost. These optimizations are solved as LP problems.

Nodes 4-6 are spawned from node 3 since it had the lowest minimum cost. In these nodes only pipe #3 remains continuous. Note that no feasible solution can be found for node 6, so it is "fathomed". After spawning nodes 10-13, node 11 represents a current best solution. Nonlinear analysis is performed at all current best solutions to check actual feasibility. Nodes 2,7, and 8 are fathomed since their minimum costs are greater than that of the current best solution. The strategy continues until all possible branches are fathomed. The solution is found at node 16. Two nonlinear analyses and one gradient analysis were performed in the process.

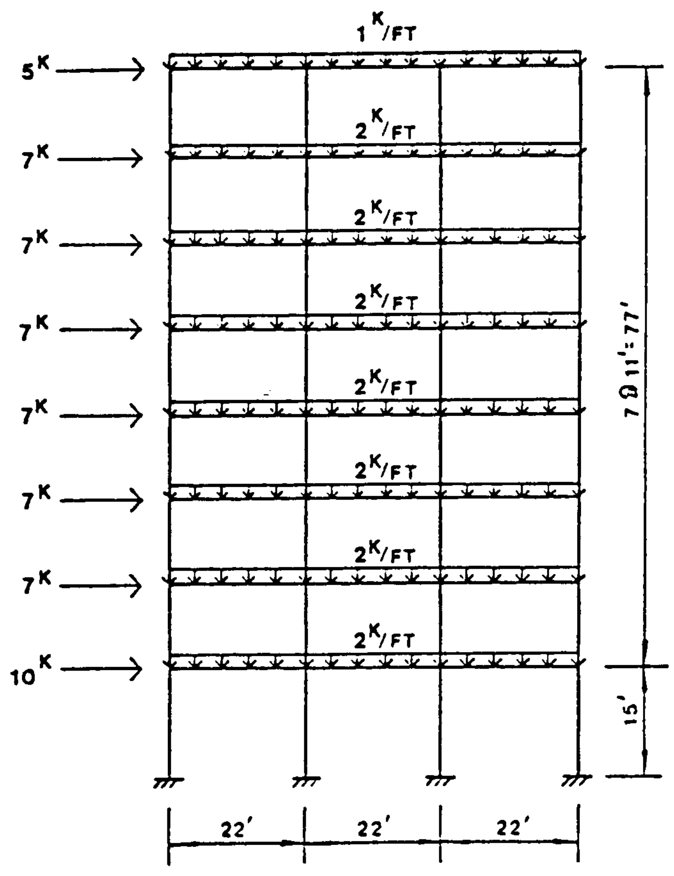


OPTIMIZATION WITH DISCRETE VARIABLES
EXAMPLE: STEEL FRAME

The problem was to select from among the 194 standardized sections published by AISC the optimal 16 sections for the frame shown. The 16 sections to be selected included 8 girders (one continuous girder for each floor) and 8 columns (interior and exterior columns where each column was continuous for two stories). The AISC combined stress constraints were imposed for each of the 56 members, and total frame weight was minimized. Only in-plane deformation was considered, the K-factor for all members was taken as 2.5, and continuous lateral support was assumed.

There were three design variables for each of the 16 sections during continuous optimization, namely: area, moment of inertia, and section modulus. 74 enveloping constraints were generated about the 194 standardized sections in 3 dimensions. Since this would add 16x74=1184 constraints to the problem, only 24 enveloping constraints were retained accounting for 80% of the area of the convex hull. This added 24x16=384 to the problem. The continuous optimum was found, and it had a weight of 36,257 lbs.

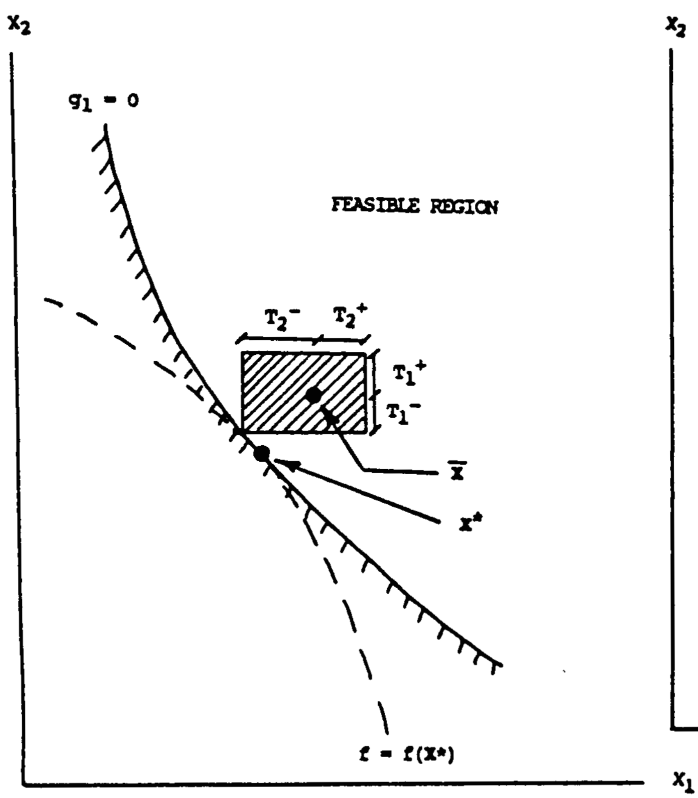
Neighborhoods about the continuous optimum were constructed such that 3 or 4 standardized sections were included for each of the 16 sections to be selected. An exhaustive search over the standardized sections in these neighborhoods would require 429,981,696 analyses. The linearized branch and bound strategy required one gradient analysis for linearization and 13 regular analyses to verify feasibility of current best solutions. The discrete optimum had a weight of 40,337 lbs. In going from the continuous optimum to the discrete optimum, all the areas increased, but 13 out of 16 moments of inertia decreased, and 6 out of 16 section moduli decreased.



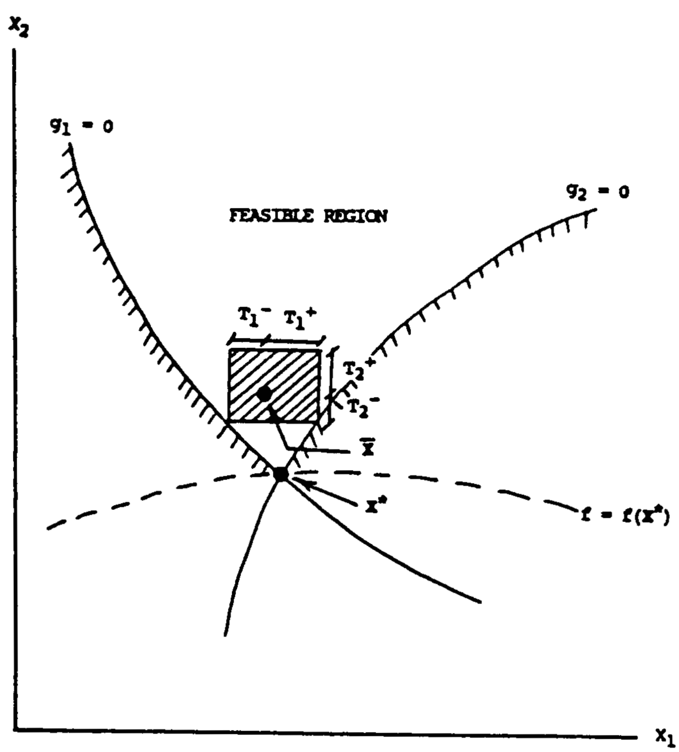
OPTIMIZATION WITH TOLERANCES
PROBLEM A: TOLERANCES SPECIFIED

In this problem, one seeks a new design as close as possible to the optimal design such that constraints are not violated for any design within the specified tolerance ranges of this new design. This means that the shaded box in the figure must be entirely feasible, and centered as close as possible to the optimum.

The problem is solved by linearizing the constraints about the optimum. Examination of the signs of the terms in the constraint gradient vectors indicate which sides of the box control for each constraint. By minimizing the square of the distance from the box center to the optimum, the problem can be solved as a quadratic-programming problem.



Problem A (nonfully constrained optimum)



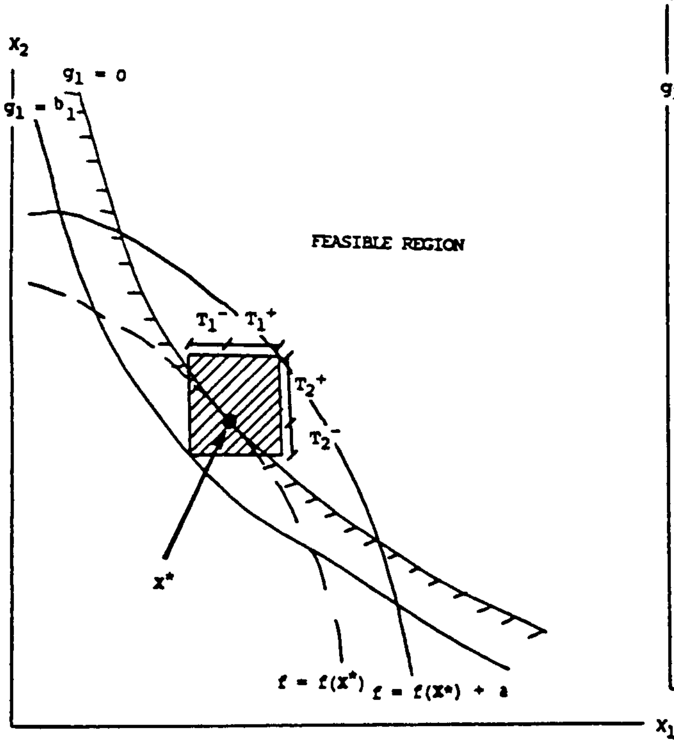
Problem A (fully constrained optimum)

OPTIMIZATION WITH TOLERANCES
PROBLEM B: TOLERANCES COMPUTED

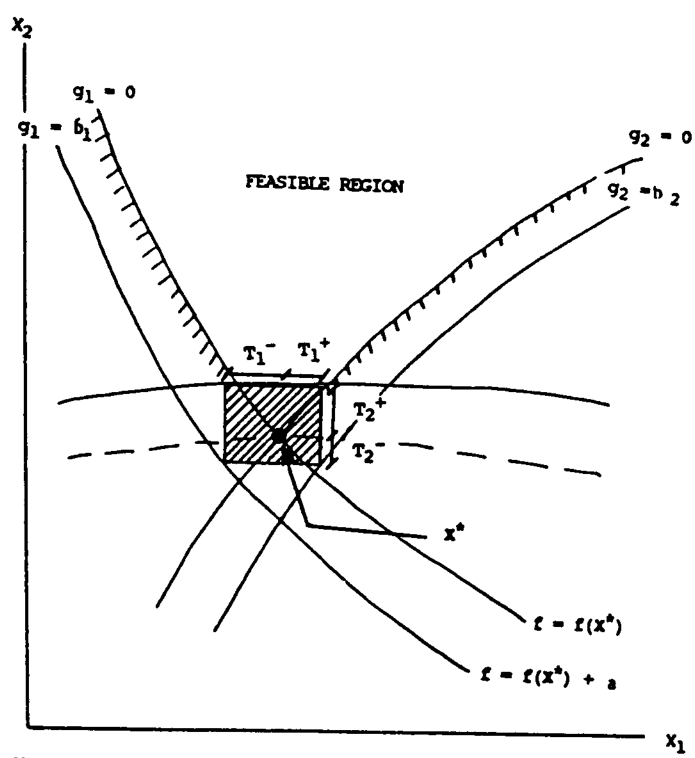
In this problem, one seeks the largest tolerance ranges according to some norm about the optimum design such that any design within these tolerance ranges does not exceed specified acceptance levels for the objective and constraints. Thus, the user "backs off" from the optimum values for the objective and constraints by specifying acceptance levels. The shaded box in the figure must be entirely contained within the region bound by these acceptance levels.

Again the objective and constraints are linearized, and the controling sides of the box are identified for each constraint. Various norms of the tolerances could be maximized. If the L1 norm is used (tolerances simply added), the problem can be solved as a linear-programming problem.

To keep the problem well-posed, upper and lower bounds on the tolerances should be specified. Lower bounds represent the user's estimate of the tolerance that he must absolutely have as a minimum for each variable. Upper bounds represent his estimate of the tolerance that, if achieved, is all he needs for a variable (at this point the optimization should try to increase tolerances in other variables).



Problem B (nonfully constrained optimum)



Problem B (fully constrained optimum)

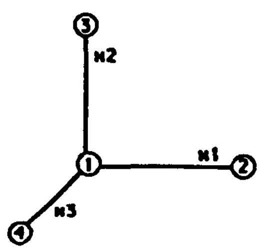
Approximation of Computationally Expensive and Noisy Functions

The approximation of design functions with first or second order polynomials for optimization has several advantages: the polynomials smooth noisy functions, which can improve algorithm performance, 2) analysis and optimization can be decoupled so that optimization can be executed on one computer and the analysis on another, and 3) the number of analyses required to reach an optimum, particularly for noisy functions, can often be significantly reduced. Approximation is also an important aspect of several problem decomposition schemes.

The approach taken in this research is to use statistical test plans to determine where analysis should be run in order to make the approximation. The statistical test plans yield approximations that can be superior to approximations made from Taylors series expansions because the analyses are spread throughout the range of the function being approximated, and, for each analysis, more than one variable is changed at a time (in contrast to finite difference derivative), making it possible to use several analyses to estimate a particular model coefficient.

This advantage is demonstrated in the figure below comparing the analyses evaluated with a "one at a time" test plan to the analyses evaluated with a saturated factorial test plan. For a problem with three variables, both strategies require a base point (variables set to -1) and three other analyses. In the "one-at-a-time" plan, each variable is perturbed in turn while the others are left at the base values. The effect of each variable can only be estimated from two analyses. In the saturated plan, however, two variables are perturbed for each analysis; as a result, twice as much information is available to estimate the model coefficient of each variable.

"one-at-a-time" or finite difference

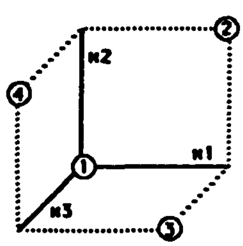


	x1	x2	x3
analysis 1	-1	-1	-1
analysis 2	+1	-1	-1
analysis 3	-1	+1	-1
analysis 4	-1	-1	+1

effect of x1 = 2 vs. 1 (2 analysis points)
effect of x2 = 3 vs. 1 (2 analysis points)
effect of x3 = 4 vs. 1 (2 analysis points)

The effect of each variable is estimated from only one other evaluation.

factorial design



	x1	x2	x3
analysis 1	-1	-1	-1
analysis 2	+1	+1	-1
analysis 3	+1	-1	+1
analysis 4	-1	+1	+1

effect of x1 = 1,4 vs. 2,3 (4 analysis points)
effect of x2 = 1,3 vs. 2,4 (4 analysis points)
effect of x3 = 1,2 vs. 3,4 (4 analysis points)

Twice as much information is available to estimate the effects of each variable for the same number of function evaluations.

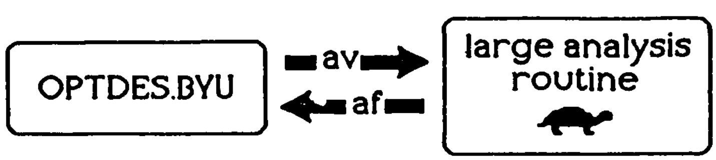
The capability to approximate functions has been integrated fully into the OPTDES package. The user first specifies variable range limits for the approximation. OPTDES will then generate a test plan within those limits and write the analysis variable values in the proper file format for the user's analysis software. After the analysis is finished, OPTDES reads the analysis results and performs regression analysis to obtain the model, displaying the model goodness of fit. The user can then optimize directly on the model.

An example of this operation is shown below. For this example, which involved large scale thermal analysis, 57 analysis calls by OPTDES were required when direct optimization was used. Using model approximation with statistical test plans to determine where analysis should be performed, the number of analysis calls was reduced to 24.

Although very efficient test plans exist for estimating models with linear coefficients, statistical plans for second order models tend to be expensive, in that they require more analyses than the number of estimated coefficients. The popular Box-Behnken plan, for example, requires 25 analyses to estimate 15 second order coefficients for a problem with four variables. These extra analyses are used in part to determine the variance, or random error, of the analysis results. In a computer model, such variance does not usually exist. The statistics department at BYU has been testing efficient second order test plans that require the same number of analyses as coefficients for use in approximation for optimization. We feel that these test plans will be very useful for this application.

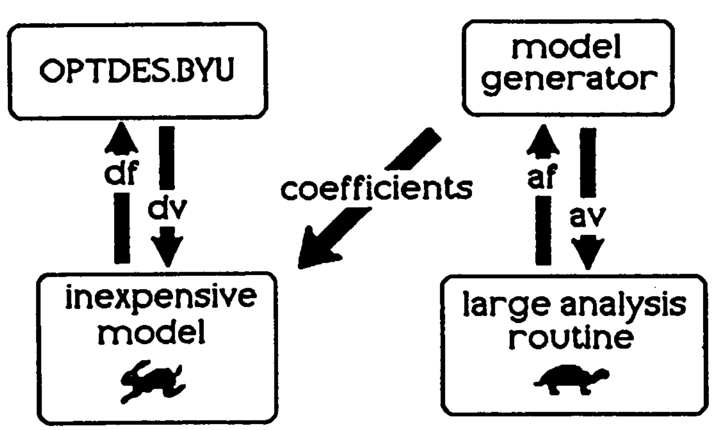
Without Approximation

57 calls



With Approximation

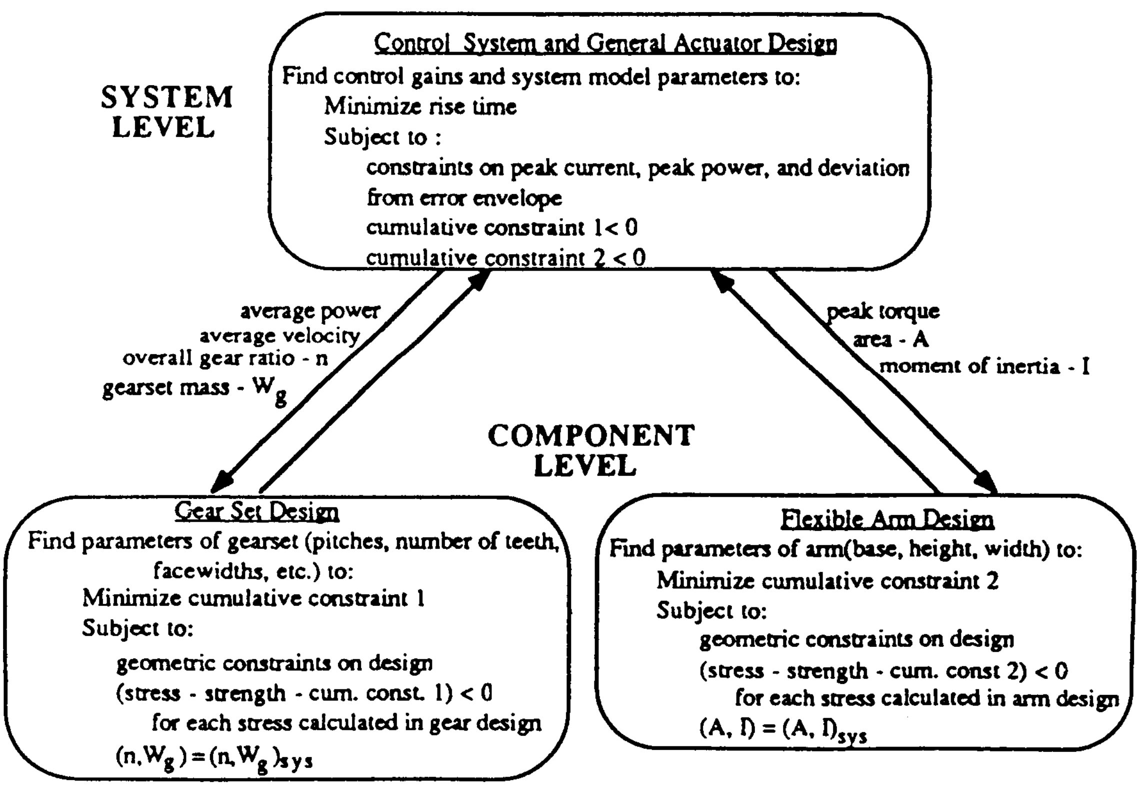
24 calls



Electromechanical Actuator/Control System Design Using Decomposition

Advancements in robotics and computer storage media require servomechanisms that are quick, precise, and powerful. The dynamic performance of a control system is ultimately limited by the actuator hardware. In addition, practical design considerations such as weight, volume, and power are dependent on actuator parameters. Normally the design of an actuator and its control system are approached sequentially: an actuator is selected or designed; the control system for the actuator is then determined. The objective of this research was to integrate the design of the actuator and control system in order to optimize the transient response. Because the design of such a system can be complex, decomposition methods were studied as a means of approaching the design problem. The discrete variable capabilities of OPTDES were also used to select an optimal motor from catalog values.

The electromechanical actuator considered consists of a permanent magnet dc motor coupled to a double reduction gear set with inline input and output shafts driving a flexible arm carrying an inertial load. The objective of the design problem was to minimize rise time of the actuator, subject to constraints on over/undershoot. The 22 design variables included six control gains, the resistance, inductance, time constant, torque constant, and rotor inertia of the motor, the detailed design of the gear set and the actuator arm. The problem was decomposed heuristically according to the physical makeup of the system, as given below.



Decomposition of Electromechanical Actuator.

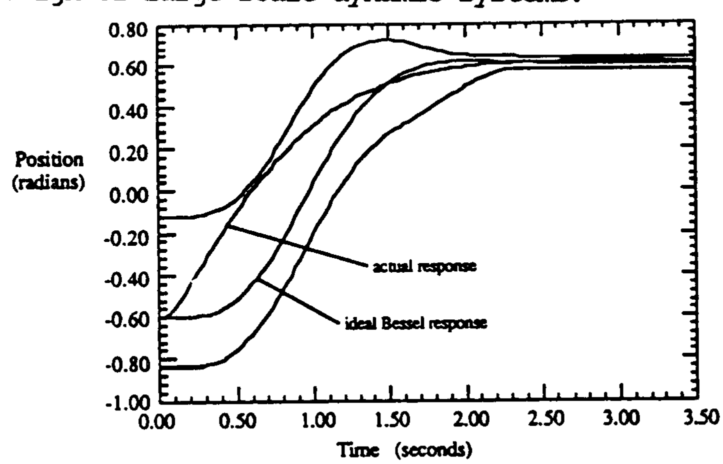
As the figure shows, the optimization of the system time response was assigned to be the overall objective of the system. The design of the gear set and actuator arm were designed at the component level. After decomposition, the system level problem contained 11 system variables; the gear component design problem had 8 design variables, and the arm design problem had 3 design variables. The strategy for solving the decomposed problem was that developed by Sobieski, using cumulative constraints. However, the cumulative constraint was not formulated using the Kresselmeier-Steinhauser function, but was formulated using the simple form,

$$\begin{array}{ll} \text{Minimize} & S \\ \text{subject to} & \text{Constraint}_i - S \leq 0 \quad \text{for all } i \end{array}$$

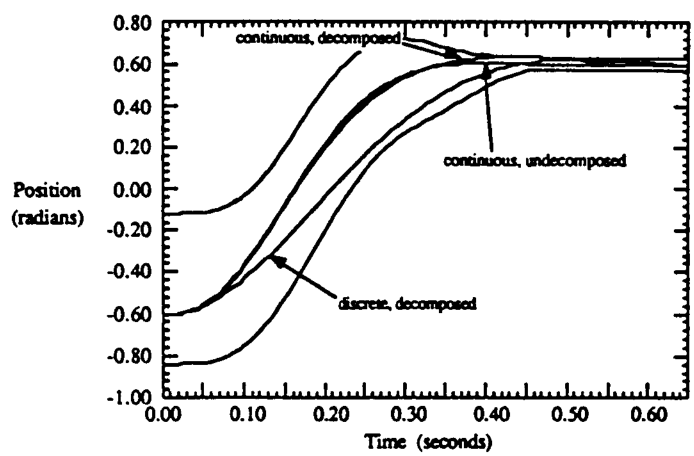
Minimizing S tends to maximize the feasibility of the design.

The step response of the system before optimization is shown below. The top and bottom response curves in the figure represent the error envelope the response must stay within. The optimal response is given in the second figure. The "continuous" response in the figure is the optimal response with the motor variables modeled as continuous variables. The "discrete" response is the response of the optimal actuator with the optimal discrete values of the motor as selected from a vendor's catalog.

The results from this sample problem show decomposition to be a potentially valuable tool in the design of large scale dynamic systems.



Electromechanical Actuator Response Before Optimization.

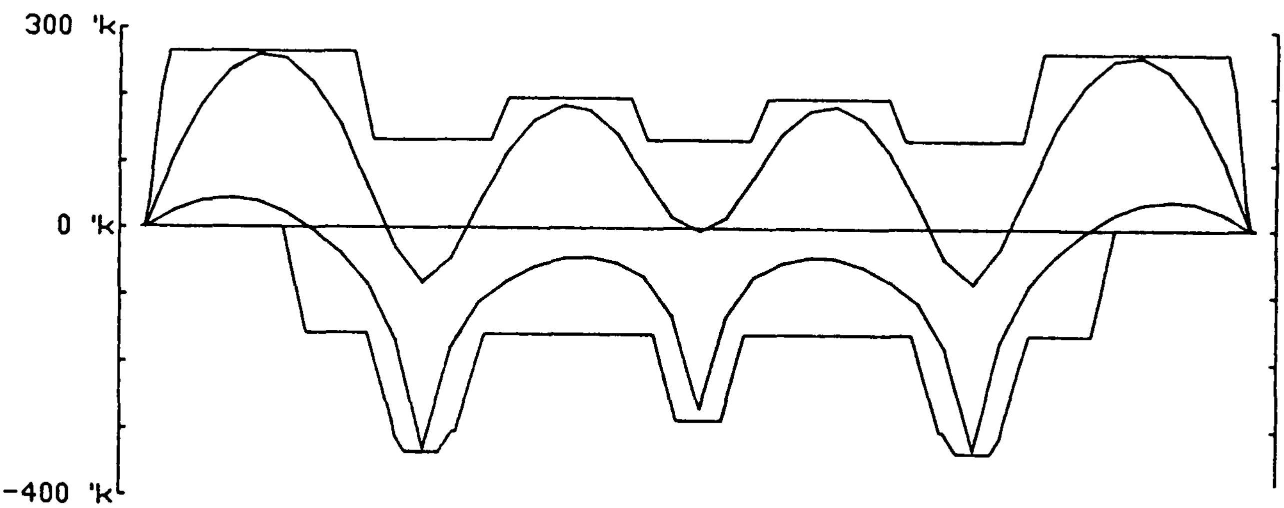


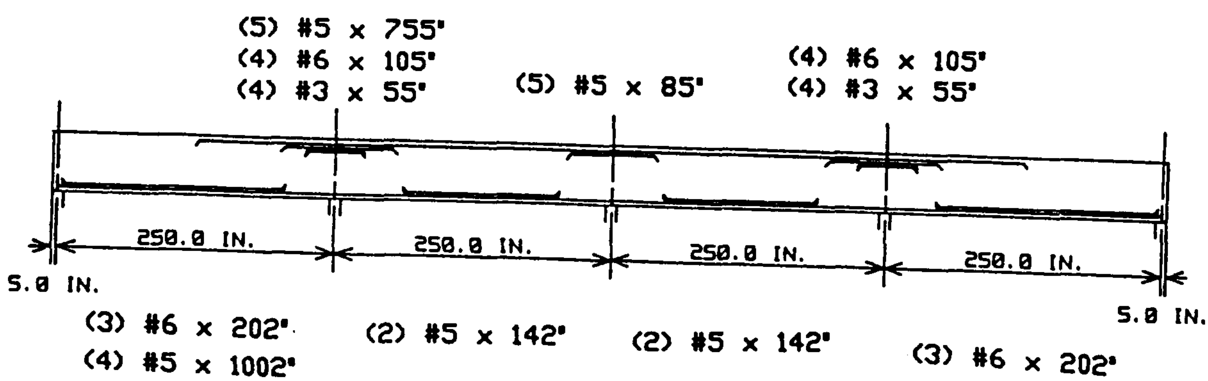
Optimal Response Plots for Electromechanical Actuator.

CONFIGURATION OR TOPOLOGICAL OPTIMIZATION
EXAMPLE: REINFORCEMENT IN A CONCRETE BEAM

The problem here was to design the reinforcement in a concrete tee-beam over four equal 250-inch spans. The material properties and dimensions of the concrete beam were given. The moment envelopes under possible loading conditions are given in the figure. All constraints imposed by the ACI 318-83 Specification were applied. The reinforcement cost was minimized, which included material cost ($.25/lb of steel) and placement cost ($1.00 per bar).

The figure shows the optimum design. Note that the number of bars (given in parentheses), the discrete bar sizes (#5, #6, #3, etc.), the bar lengths (in inches), and the locations (top or bottom, over support or in the middle of the span, and layers) are given for each group of bars. The optimal moment capacity envelope is also shown in the figure.





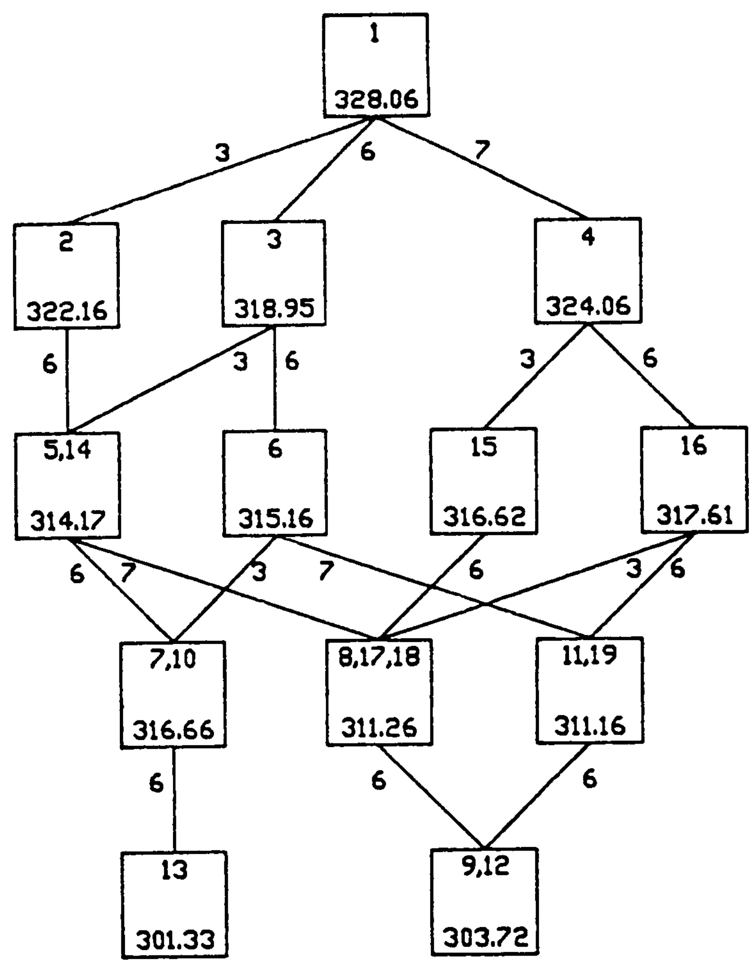
CONFIGURATION OR TOPOLOGICAL OPTIMIZATION
RULE-BASED SEARCHING STRATEGY

The problem is solved by generating reinforcement configurations. The number of bar groups with the location and number of bars for each group are needed to specify a configuration. The configuration is then optimized via the linearized branch-and-bound method to determine the bar sizes, bar lengths, and total reinforcement cost. The figure shows boxes representing the various configurations. The boxes contain the configuration numbers according to the order in which they were generated, and the total reinforcement cost.

A rule-based algorithm is used to search through configurations. Configuration #1 contains both positive and negative "primary" groups extending all the way across the beam with enough steel to satisfy the minimum reinforcement requirements. "Secondary" groups are included over each support and in each span to provide necessary moment capacity.

"Child" configurations are spawned from the "parent" configurations as shown in the figure according to one of eight heuristic rules. These rules govern the deletion/addition of a bar within a group or the deletion/addition of an entire bar group. For example, rule #3 states that if the number of bars in a secondary group is more than twice as many as the number of bars in the primary group, divide the secondary group into two groups. The algorithm stops when no more rules apply to any children.

Note that the same child may be spawned from more than one parent. Note also that the cost of configuration 7,10 was greater than the cost of configurations 5,14 or 6 from whence it came. Nevertheless configuration 7,10 spawned configuration 13 which was the eventual optimum.



Application of Knowledge-Based Systems and Optimization for the Design of a Valve Anti-Cavitation Device

This research involves the design of a device to control cavitation for liquid valves. Cavitation can cause erosion of valve material and premature valve failure. An approach for preventing cavitation is to force the liquid through a series of expansion holes and contraction channels, machined into concentric cylinders, as shown in the figure below. The cylinders together comprise the "anti-cavitation retainer." A local valve company desired to develop software to automate the design of the retainer. Design of a good retainer can be complex and requires an experienced engineer.

Initially expert system technology was applied to capture the design rules of the expert. However, it became apparent as the expert described his design procedure, that many of his rules were associated with how to change variables to obtain a good design. These rules were replaced with an optimization algorithm.

The package that was developed consisted of a small expert system which applied the true heuristic rules to the problem, setup the optimization problem, called the algorithm, and interpreted the results. The optimization algorithm determined the values of variables. This strategy of combining heuristic search with numerical search could apply to a broad spectrum of engineering design problems. Knowledge-based systems and numerical optimization are complementary approaches that span both the qualitative and quantitative aspects of design.

When completed, the software was tested on ten actual design problems that had been previously solved by the expert. The expert verified the adequacy of the designs produced by the package. In five cases, the software developed satisfactory designs with a fewer number of cylinders--these designs would be cheaper to produce. In two cases, the package produced designs that violated a fewer number of customer requirements. The remaining three designs were equally satisfactory.

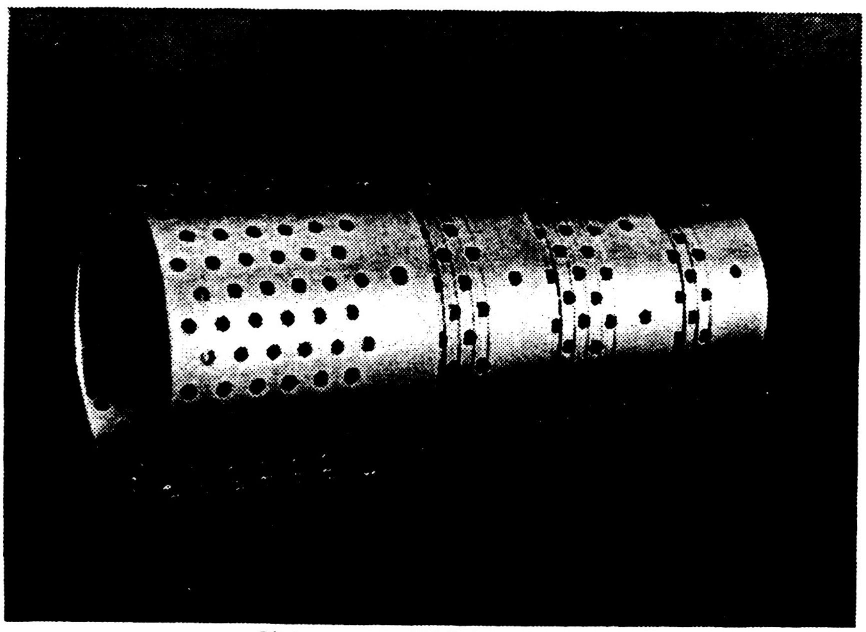


Photo courtesy of Valtek Incorporated.
Anti-Cavitation Retainer

REFERENCES

OPTDES.BYU

A.R.Parkinson, R.J.Balling, and J.C.Free, "Exploring Design Space: Optimization as Synthesizer of Design and Analysis", COMPUTERS IN MECHANICAL ENGINEERING, Vol. 3, No. 1, Pp. 28-36, 1985

A.R.Parkinson, R.J.Balling, and J.C.Free, "OPTDES.BYU: A Software System for Optimal Engineering Design", ASME COMPUTERS IN MECHANICAL ENGINEERING CONFERENCE, August 12-16, 1984, Las Vegas, NV

OPTIMIZATION WITH DISCRETE VARIABLES

K.Hager and R.J.Balling, "A New Approach for Discrete Structural Optimization", ASCE JOURNAL OF STRUCTURAL ENGINEERING, Vol. 114, No. 5, Pp. 1120-1134, May 1988

M.W.Brown and R.J.Balling, "Discrete Optimization of Water Networks", ASCE 5TH CONFERENCE ON COMPUTING IN CIVIL ENGINEERING, March 29-31, 1988, Alexandria, VA

OPTIMIZATION WITH TOLERANCES

R.J.Balling, J.C.Free, and A.R.Parkinson, "Consideration of Worst-Case Manufacturing Tolerances in Design Optimization", ASME JOURNAL OF MECHANISMS, TRANSMISSIONS, AND AUTOMATION IN DESIGN, Vol. 108, No. 4, Pp. 438-441, December, 1986

APPROXIMATIONS BASED ON STATISTICAL TEST PLANS

J.W.Free, A.R.Parkinson, G.R.Bryce, and R.J.Balling, "Approximation of Computationally Expensive and Noisy Functions for Constrained Nonlinear Optimization", ASME JOURNAL OF MECHANISMS, TRANSMISSIONS, AND AUTOMATION IN DESIGN, Vol. 109, No. 4, Pp. 528-532, December, 1987

DECOMPOSITION OF OPTIMIZATION

T.W.McLain, J.C.Free, and A.Teng, "Optimal Design of Complex Systems Using Decomposition Methods: Part II Example Application to Electromechanical Control", ASME 14TH DESIGN AUTOMATION CONFERENCE, September 24-28, 1988, Orlando, FL

A.R.Parkinson, R.J.Balling, A.Wu, and J.C.Free, "A General Strategy for Decomposing Large Design Problems Based on Optimization and Statistical Test Plans", INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, August 17-20, 1987, Boston, MA

TOPOLOGICAL OPTIMIZATION VIA ARTIFICIAL INTELLIGENCE

R.Ellsworth, A.R.Parkinson, and F.Cain, "The Complimentary Nature of Symbolic Computing and Numerical Optimization in Engineering Design Software", ASME 13TH DESIGN AUTOMATION CONFERENCE, September, 1987, Boston, MA