

IN-05
167899

NASA Technical Memorandum 107761

P-13

**MULTIDISCIPLINARY DESIGN OPTIMIZATION:
AN EMERGING NEW ENGINEERING
DISCIPLINE**

Jaroslav Sobieszczanski-Sobieski

(NASA-TM-107761) MULTIDISCIPLINARY
DESIGN OPTIMIZATION: AN EMERGING
NEW ENGINEERING DISCIPLINE (NASA)
13 p

N93-27258

Unclas

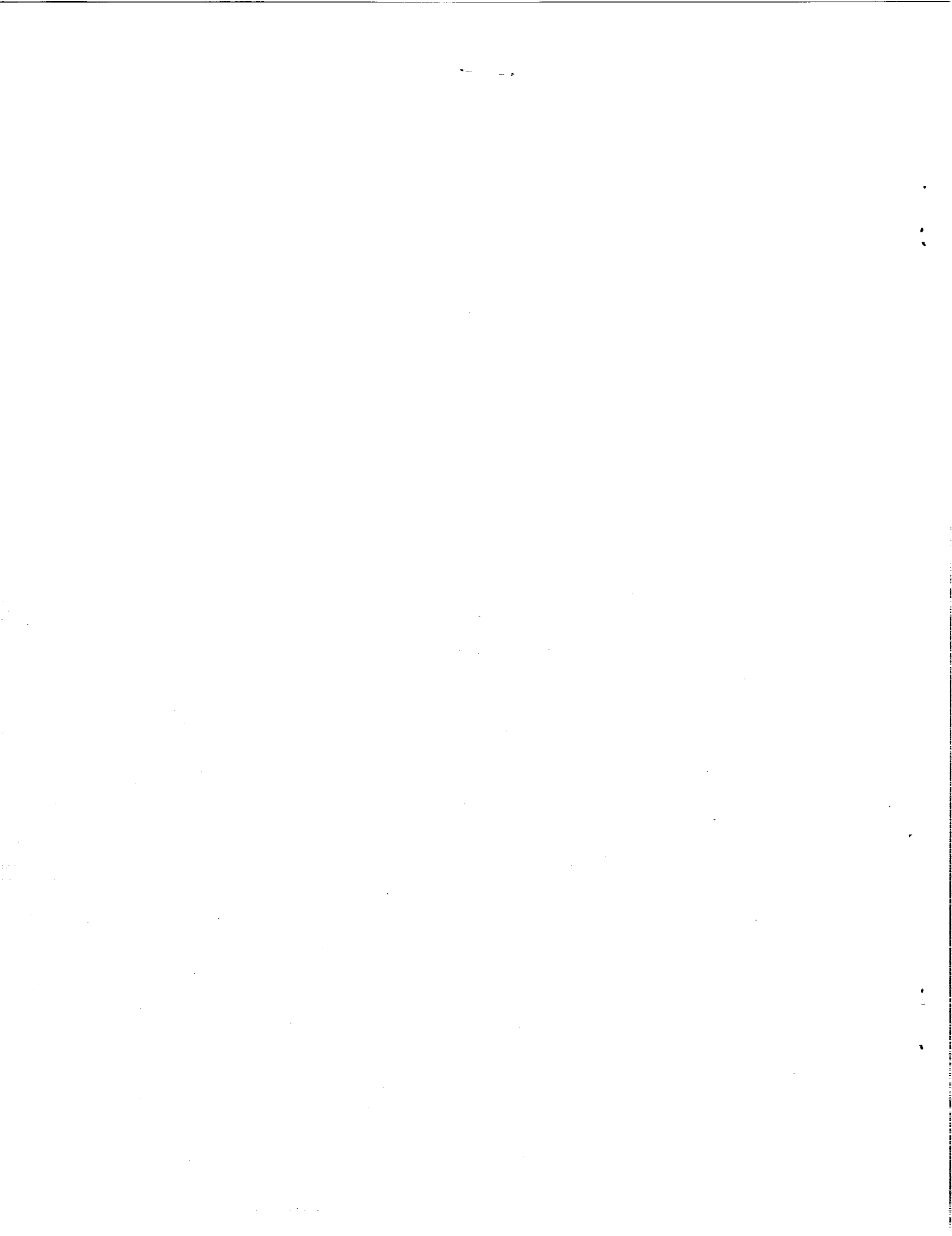
G3/05 0167899

May 1993



National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-0001



MULTIDISCIPLINARY DESIGN OPTIMIZATION: AN EMERGING NEW ENGINEERING DISCIPLINE

by

Jaroslav Sobieszczanski-Sobieski
NASA Langley Research Center
Hampton, Virginia

ABSTRACT

This paper defines the Multidisciplinary Design Optimization (MDO) as a new field of research endeavor and as an aid in the design of engineering systems. It examines the MDO conceptual components in relation to each other and defines their functions.

INTRODUCTION

During the last decade, a set of previously disjointed ideas and tools has crystallized into a new technology with a cohesiveness and distinct character that give it characteristics of a new engineering discipline. This discipline is unique in its role in engineering design where it acts as an agent binding together the other engineering disciplines involved. The new discipline is often called the Multidisciplinary Design Optimization (MDO); e.g., Ref. 1, and will be referred to by this name herein. Alternative names such as Multidisciplinary Analysis and Optimization (MAO) and Multidisciplinary Design Methodology or Technology, (MDM or MDT) have also been proposed.

The purpose of this paper is to introduce the definition of MDO and of its principal conceptual components which are Design-Oriented Analysis, Approximation Concepts, System Mathematical Modeling, Design Space Search, Optimization Procedure, and Human Interface as shown in Fig. 1. The paper examines functionality and mutual relationships of these components and examines their internal structure also depicted in Fig. 1. Without attempting a comprehensive survey, selected references will be quoted to support the above discussion.

DEFINITION OF MDO

The term "methodology" is defined by Webster's Dictionary as "a body of methods, procedures, working concepts, and postulates, etc." Consistent with that generic definition, MDO can be described as a methodology for the design of complex engineering systems that are governed by mutually interacting physical phenomena and made up of distinct interacting

subsystems. The MDO methodology exploits the state of the art in each contributing engineering discipline and emphasizes the synergism of the disciplines and subsystems.

An aircraft, a car, a ship, or an electric power generation plant are all good examples of engineering systems. If one were to name their single common characteristic, it is that in their design everything influences everything else.

MDO COMPONENTS

Several conceptual components coalesced to form MDO. They are listed and briefly characterized in this section of the paper. They form the top layer in the diagram in Fig. 1 that also shows their internal breakdown which is also examined in this section. Referring to Fig. 1 often as the discussion unfolds will help in developing a broad perspective on the subject.

Design-Oriented Analysis

Designers often need to get rough solution estimates quickly and inexpensively, so typically they apply analysis repetitively to answer the "what if" questions. They also need to know sensitivity of the solution to design changes. Furthermore, in a multidisciplinary system, disciplinary analyses must interact, and the computing process produces an enormous volume of data that must be interpreted and saved to build an easily accessible data base.

A new brand of analysis has been developing in the last two decades, called the Design-Oriented Analysis to meet the needs mentioned above. Its basic features are summarized below.

Sensitivity analysis (SA). An important concern in design is about the "what if" questions. Derivatives of the output variables (state or behavior variables) with respect to (w.r.t.) the input variables (design or decision variables) constitute a precise measure of sensitivity and quantify answers to the above questions. They are referred to as sensitivity derivatives (SD's) or sensitivity coefficients (SC's).

Finite differencing (FD) on an analysis code is one way to calculate SD. However, the FD approach has shortcomings. Its accuracy deteriorates with the step length in nonlinear problems, but making the step too short may incur excessive truncation errors (Ref. 2 offers a technique for optimal setting of the step length). If analysis is iterative, then an FD executed at the nominal solution may produce meaningless results (a remedy is given in Ref. 3). Finally, the FD cost grows linearly with the number of the design variables; hence it becomes excessive in large problems, even if the inexpensive reanalysis techniques discussed later in the paper are used.

Motivated by the above shortcomings, several alternative techniques have evolved. Because in the field of engineering analytical closed-form solutions amenable to symbolic differentiation are usually not available, a quasi-analytical approach based on the Implicit Function Theorem is one popular technique. When applied to a disciplinary analysis, the technique differentiates the governing equations of the analysis to obtain the companion sensitivity equations that,

according to the above theorem, are always linear, simultaneous algebraic equations in which the SD's appear as unknowns. The above approach results in two versions of SA. The direct version generates SD's of all the analysis unknowns. If one is interested in the derivatives of only a subset of the unknowns, then an alternative referred to as the adjoint variable method is in order.

The SA methodology has been most fully developed in computational chemistry, control theory, and in structural analysis. In the latter the capability to obtain derivatives of displacement, stress, and vibration frequency w.r.t. cross-sectional dimensions and shape variables is now routinely available. Reviews of the subject may be found in Refs. 4, 5, and 6. Similar developments are currently under way in aerodynamic analysis; e.g., Refs. 7 and 8, and the beginning is being made in other engineering disciplines as well.

To avoid the costs of new coding required when quasi-analytical techniques are retrofitted in the existing codes, an "automatic differentiation" method has recently become available. The method applies a line-by-line symbolic differentiation to an existing code and stores numerical values of the dependent variables for each code line. Moving from one line to the next, the algorithm links the derivatives in a chain-differentiation manner as required by the variable dependencies from the beginning to the end of the code. The result is a set of the derivatives of the output w.r.t. input. The method is implemented in the form of an automatic differentiator code that reads the user's existing source code and produces a new source code that retains exactly the same capability as the original but is enhanced with an option of computing the SD's. A survey of the differentiator codes currently available is given in Ref. 9, and examples of engineering applications may be found in Refs. 10 and 11.

Typically, the above SA techniques apply to a code representative of an engineering discipline. In a complete vehicle analysis, these codes form a system coupled by the output-to-input data cross-flow that simulates behavior of the entire vehicle system. Since neither of the above SA techniques is practical in a direct application to a modular system of codes, a special technique for such systems has been developed (Refs. 12 and 13). That technique, called the system SA, uses the SD's obtained for the system modules by any of the above SA methods as coefficients to build a set of the linear, algebraic equations also derived from the Implicit Function Theorem. These equations yield the SD's of the system behavior with respect to the design variables, with the interactions among the disciplines accounted for. Examples of applications of the above technique are discussed in Ref. 14.

The SD's may be used directly to support engineering judgment and intuition, or they may be supplied to a design space search algorithm to conduct a formal optimization. Implications of these SD uses to the design process are discussed in Ref. 15.

Often times, a "what if" question arises at the end of an optimization process in which certain quantities, e.g., structural material stress allowable, were held constant. The question is about the influence of these quantities (optimization parameter, or OP) on the objective and on the

design variables at the constrained optimum. Instead of repeating the optimization with an OP perturbed, one may obtain an answer in the form of the derivatives of the objective and of the optimal design variables w.r.t. OP using an algorithm described in Ref. 16.

Inexpensive reanalysis. The objective here is to make the analysis code organization flexible enough so that a new solution reflecting a change to the input variables may be obtained by repeating as small a subset of the analysis as possible. To achieve this in a rigidly structured code that offers a menu of different execution options set a priori in anticipation of the "what if" questions that user might be asking (where each question is defined by a subset of the input variables that might be changed) is very difficult. It is practically impossible to anticipate all such questions that might arise in the operation of a code with input that is diverse and voluminous. Even if a perfect anticipation was possible, a combinatorial explosion of the number of the execution options would set in.

One remedy is to organize a program as a system of modules and to establish the data dependence information that for each module input identifies a source either in the output of another module or in the input from outside of the system. That information, tabulated and recorded as a part of the system, enables the system executive module to find out the following: 1.) what modules and output data will be affected and, therefore, what modules will have to be re-executed when there is a change to particular input data (a forward chaining mode), or 2.) to determine what modules must be executed to compute particular output data (a backward chaining mode).

To satisfy a user's request for particular output data, the stored output (output from each execution is routinely saved) is searched first to see if the requested data are readily available there. If not, the executive module scans the data dependence data and the stored output to determine the smallest subset of modules and its execution sequence which will produce the required data, and then, commands the data calculation. If the user changes the particular input data, the executive module activates a computational sequence which will produce the data that are affected by the change. In either case, the user request is satisfied with a minimum of a computational effort. This type of a system is called nonprocedural because the user does not have to choose among the execution options or to code the execution procedure appropriate to the requests. A pioneering example of such a non-procedural system may be found in Ref. 17.

Computational cost-accuracy trade. An example of a technique that enables one to trade the solution accuracy for its computational cost in repetitive applications is given in Ref. 18. This technique uses two mathematical models of the same physical phenomenon: 1.) a refined model (R), costly to analyze and invoked sparingly; and 2.) a simplified one, (S), inexpensive and used often. The ratio of an element of the R output to the corresponding ratio of the S output is termed a correlation factor and those factors form a correlation vector. Assuming that the same design variables govern both models, derivatives of the above vector w.r.t. these variables may be computed in terms of the SD's of the R and S outputs w.r.t. the same variables. Then, the

state of R for modified design variables may be approximated by analyzing S and multiplying its output vector by the correlation vector updated by the derivative-based linear extrapolation. Periodically, the R model has to be reanalyzed to recover from the extrapolation errors. As demonstrated in Ref. 18, the accuracy of such extrapolation is better than the one obtained with constant correlation factors.

Data management and visualization. Because of its repetitive nature, the Design-Oriented Analysis produces a very large volume of data. Some of these data has to be stored for future use, some of it must be immediately transferred among the modules, and some of it should be made available to the user for interpretation. The data storage requires a data base facility, preferably based on the relational storage scheme supporting the data recall by attributes. However, the overhead required by such a storage facility may be unacceptable for data transfers that have to be re-executed often, e.g., in an iterative loop; hence a separate, direct data transmission mechanism may be superior for that purpose. An example of such a two-tier data handling system is described in Ref. 19.

To support the user judgment, intuition, and the continuity of the train of thought in design, it is imperative to present the data visually. The use of color as the additional dimension has begun to be regarded as indispensable.

Approximation Concepts

Direct coupling of the design space search code (DSSC) to the analysis is impractical when the analysis is expensive because the number of calls from DSSC to analysis for new information may be excessive. To gain control over that number, it became customary in large applications to refer the DSSC calls to an approximate analysis (AA), while invoking the full analysis infrequently at the rate required by the approximation error control. The approximation concepts underlying AA range from the Taylor series approximations based on the SD's (Ref. 20) to the Design of Experiments with a fitted response surface (Ref. 21). This class of methods has recently been enhanced by the use of orthogonal arrays (Ref. 22) and the application of the neural nets in lieu of the fitted response surface (Ref. 23).

System Mathematical Modeling

It is axiomatic that mathematical model of an engineering system is a modular system of codes, rather than a monolithic code. Because codes in a modular software system send data to each other, the overall computational efficiency will benefit from having the mathematical models embodied in those codes set up to minimize the additional data processing required for the data transfers among the codes. An example of this is an aircraft wing design where the aerodynamic analysis that uses a 3D grid interacts with a finite element model that uses its own nodal mesh. Ideally, the aerodynamic analysis module should output forces at the structural mesh nodes, and the structural analysis module should send the displacements at the surface aerodynamic grid points. Furthermore, to support the shape optimization both grids should be

parametrized in terms of the shape variables of interest so that when the wing shape changes both grids adjust without having to be regenerated from scratch.

When mathematical models of two or more disciplines interact very often in a multidisciplinary system analysis and send large volumes of data to each other, they may be candidates for merging at the equation level. An example of this is recently developed heat transfer analysis blended with structural analysis using a shared finite element model (Ref. 24).

Driven by the increasing concern of cost as the dominant consideration in design, the notion of mathematical modeling has recently begun to extend from the design phase (concerned primarily with physical phenomena) to the phases of the product specification development, manufacture, and operation (dominated by man-made, nonphysical processes). Once these four principal phases of the product life are mathematically modeled at the same level of fidelity now common in the design phase and their couplings are accounted for, the entire life cycle will become amenable to optimization for minimal total cost as forecast in Ref. 25.

Decomposition

In the time-honored "divide-and-conquer" approach to large engineering tasks, a number of techniques have been developed to partition large engineering design optimization tasks into smaller tasks that remain coordinated so that their coupling and therefore their synergism are not lost. The systems amenable to decomposition are usually categorized as hierarchic, non hierarchic, and hybrid. The distinguishing feature is the data flow among the modules that collectively simulate the system. In the hierarchic system, the modules form a pyramid with the data flow starting from the top, so that several "children" modules may receive input data from the same "parent" module; however, the children modules do not communicate directly with each other. In contrast with that, the modules in a non-hierarchic system may communicate with each other without any restrictions so that one cannot identify a set of children to a parent. A mixture of the two categories constitutes a hybrid system.

It is often obvious at a glance how to organize a given collection of modules into a system of one of the above categories on the basis of the examination of the data flow among the modules. For large collections of modules, this may be aided by formal means such as the N-square method embodied in a code described in Ref. 26. Examples of engineering system optimization based on decomposition may be found in Refs. 27 and 28 for a hierarchic case, and in Refs. 29 and 30 for the non-hierarchic cases.

Design Space Search

The purpose of the search is to produce a design improvement and, ultimately, to find a constrained minimum. A very large variety of the DSSCs (search algorithms, optimizers) performing that function became available because of vigorous research and development in Operations Research. A survey may be found in Ref. 31. An alternative approach to locating

a constrained or unconstrained minimum in a design space known as the optimality criteria method has been reviewed in Refs. 32 and 33.

Optimization Procedure

Contrary to a common misconception, an optimizer that searches the design space is only one of many components of an optimization procedure. A typical such procedure ties together the MDO elements discussed in this paper in an execution sequence. A flowchart of an optimization procedure for non-hierarchic systems is depicted in Fig. 2 and an example of its application may be found in Ref. 14. Many different procedure organizations are possible depending on the application and the computer software and hardware available. A number of different schemes are discussed in Ref. 34 for a serial computer implementation. Opportunities for concurrent processing offered by the heterogeneous, massively parallel computing technology will undoubtedly spur development of new optimization procedures.

Human Interface

The MDO is definitely not a "pushbutton" approach to design; hence it relies heavily on human participation. To facilitate that participation an MDO software system typically incorporates the means for the user to review and judge the intermediate results, to intervene and override the algorithmic decisions, to reformulate the problem, and to decide when to stop. Artificial intelligence tools such as the expert systems often are found useful here.

CONCLUDING REMARKS

Multidisciplinary Design Optimization (MDO) has evolved into a methodology comprising a number of components. The principal ones have been identified as Design-Oriented Analysis, Approximation Concepts, System Mathematical Modeling, Design Space Search, Optimization Procedure, and Human Interface. These components were examined, including their breakdown into subcomponents and their relationships in the MDO methodology framework. As indicated in a review in Ref. 35, the application experience with MDO is accumulating and showing promise of significant pay-offs. In its rapidly growing maturity and diversity, and in its offering numerous opportunities for future research, MDO exhibits the attributes of a new discipline whose definition is now evolving.

REFERENCES

1. Current State of the Art in Multidisciplinary Design Optimization; An AIAA White Paper; September 1991, Washington, D.C.
2. Hull, D. G.: Numerical Derivatives for Parameter Optimization; *J. Guidance and Control*, Vol. 2, No. 2, March-April 1979, pp.158-160.
3. Haftka, R. T., "Sensitivity Calculations for Iteratively Solved Problems," *International Journal for Numerical Methods in Engineering*, Vol. 21, 1985, pp. 1535-1546.

4. Adelman, H. A.; and Haftka, R. T.: Sensitivity Analysis of Discrete Structural Systems. *AIAA J.*, Vol. 24, No. 5, May 1986, pp. 823–832.
5. Haftka, R. T.; and Adelman, H. M.: Recent Developments in Structural Sensitivity Analysis. *Structural Optimization*, 1, 1989, pp. 137–151.
6. Proceedings of the Symposium on Sensitivity Analysis in Engineering, NASA Langley Research Center, Hampton, VA, Sept. 1986; Adelman, H. M.; and Haftka, R. T.—editors. NASA CP-2457, 1987.
7. Baysal, O., and Eleshaky, M. E.; Aerodynamic Design Optimization Using Sensitivity Analysis and Computational Fluid Dynamics. AIAA Paper 91-0471, AIAA 29th Aerospace Sciences Meeting and Exhibit, Reno, NV, Jan. 1991.
8. Newman, P. A.; Hou, G. J.-W.; Jones, H. E.; Taylor, A. C. III; and Korivi, V. M.; Observations on Computational Methodologies for Use in Large Scale Gradient Based Multidisciplinary Design Incorporating Advanced CFD Codes. NASA TM-104206, NASA Langley Research Center, February, 1992.
9. Griewank, A.: The Chain Rule Revisited in Scientific Computing. *SIAM News*, Part 1, May 1991, pp. 20–21, part 2, July 1990, pp. 9–24.
10. Barthelemy, J.-F. M.; and Hall, L. E.: Automatic Differentiation as a Tool for Engineering Design. 4th AIAA/Air Force/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, Cleveland, Ohio, September 1992. Proceedings of.
11. Green, L. L.; Newman P. E.; Haigler, C. Y.: Sensitivity Derivatives for Advance CFD Algorithm and Viscous Modelling Parameters Via Automatic Differentiation. AIAA Paper 93-3321, 11th AIAA Computational Fluid Dynamic Conference, July 1993, Orlando, FL.
12. Sobieszczanski-Sobieski, J.: On the Sensitivity of Complex, Internally Coupled Systems; AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, VA, April 1988. AIAA Paper No. CP-88-2378, and *AIAA J.*, Vol. 28, No. 1, Jan. 1990, also published as NASA TM-100537, January 1988.
13. Sobieszczanski-Sobieski, J.: Higher Order Sensitivity Analysis of Complex, Coupled Systems. *AIAA Journal*, Vol. 28, No. 4, Apr. 1990.
14. Barthelemy, J-F.; Wrenn, G.; Dovi, A.; and Coen, P.: Integrating Aerodynamics and Structures in the Minimum Weight Design of a Supersonic Transport Wing. AIAA 92-2372, Presented at AIAA/ASME/ASCE/AHS/ASC 33rd Structures, Structural Dynamics, and Materials Conference, Dallas, TX, April 1992, Proceedings of.
15. Sobieszczanski-Sobieski, J.: Sensitivity Analysis and Multidisciplinary Optimization for Aircraft Design: Recent Advances and Results. Int'l Council for Aeronautical Sc., 16th Congress, Jerusalem, Aug.–Sept., 1988; Proceedings of, Vol. 2, pp. 953–964.
16. Sobieszczanski-Sobieski, J.: Barthelemy, J. F.; and Riley, K. M.: Sensitivity of Optimum Solutions to Problem Parameters. *AIAA J.*, Vol. 21, Sept. 1992, pp. 1291–1299.
17. Gage, P.; and Kroo, I.: Development of the Quasi-Procedural Method for Use in Aircraft Configuration Optimization. AIAA 92-4693, 4th AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, September 1992, Cleveland, Ohio.
18. Chang, K. J.; Haftka, R. T.; Giles, G. L.; and Kao, P.-J.: Sensitivity Based Scaling for Correlating Structural Response from Different Analytical Models. AIAA Paper No. 91-925. Presented at AIAA/ASME/ASCE/AHS/ASC 32nd Structures, Structural Dynamics, and Materials Conference, Baltimore, MD, 1991; Proceedings of.
19. Dovi, A. R.: ISSYS—An Integrated Synergistic Synthesis System, NASA CR 159221, February 1980.
20. Barthelemy, J.-F. M.; and Haftka, R. T.: Approximation Concepts for Optimum Structural Design—A Review, *Structural Optimization*, 5, 1993, pp. 129–144.

21. Montgomery, D. C.: *Design and Analysis of Experiments*. Published by John Wiley & Sons, 1991.
22. Stanley, D.; Unal, R.; and Joyner, R.: *Application of Taguchi Methods to Dual Mixture Ratio Propulsion System Optimization for SSTO Vehicles*. AIAA Paper 92-0213, AIAA 30th Aerospace Sciences Meeting and Exhibit, Reno, NV, January 1992.
23. Hajela, P.; and Berke, L.: *Neural Networks in Structural Analysis and Design: An Overview*. *Int'l J. for Computing Systems in Engineering*, Vol. 3, No. 1-4, 1992, pp. 525-539.
24. Thornton, E. A.: *Thermal Structures: Four Decades of Progress*. *Journal of Aircraft*, No. 29, 1992, pp. 485-498.
25. Tulinius, J.: *Multidisciplinary Optimization is Key to Integrated Product Development Process*. 4th AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, September 1992, Cleveland, Ohio.
26. Rogers, J. L.: *A Knowledge-Based Tool for Multilevel Decomposition of a Complex Design Problem*; NASA TP-2903, 1989.
27. Weisshaar, T. A.; Newsom, J. R.; Gilbert, M. G.; and Zeiler, T. A.: *Integrated Structure/Control Design-Present Methodology and Future Opportunities*. ICAS-86-4.8.1, Sept. 1986.
28. Wrenn, G. A.; and Dovi, A. R.: *Multilevel Decomposition Approach to the Preliminary Sizing of a Transport Aircraft Wing*. *AIAA Journal of Aircraft*, Vol. 25, No. 7, July 1988, pp. 632-638.
29. Unger, E. R.; Hutchinson, M.G. ; Rais-Rohani, M.; Haftka, R. T.; and Grossman, B.: *Variable-Complexity Multidisciplinary Design of a Transport Wing*. *Int'l J. of Systems Automation: Research and Applications (SARA) 2*, 1992, pp. 87-113.
30. Coen, P. G.; Sobieszczanski-Sobieski, J.; and Dollyhigh, S. M.: *Preliminary Results from the High-Speed Airframe Integration Research Project*. AIAA Paper No. 92-1004; 1992 Aerospace Design Conference, February 1992, Irvine, CA.
31. Arora, J. S.: *Introduction to Optimum Design*, McGraw-Hill, New York, 1989.
32. Venkayya, V. B., "Optimality Criteria: A Basis for Multidisciplinary Optimization," *Computational Mechanics*, Vol. 5, pp. 1-21, 1989.
33. Berke, L., and Khot, N. S., "Structural Optimization Using Optimality Criteria," *Computer Aided Structural Design: Structural and Mechanical Systems* (C.A. Mota Soares, Editor), Springer Verlag, 1987.
34. Sobieszczanski-Sobieski, J.: *From a Black-Box to a Programming System*, Ch. 11 in *Foundations for Structural Optimization—A Unified Approach*; Morris, A. J., ed. J.; Wiley & Sons, 1982.
35. Sobieszczanski-Sobieski, J.: *Aircraft Optimization by a System Approach: Achievements and Trends*, International Council of Aeronautical Sciences, 18th Congress, Beijing, China, September 1982, Proceedings of.

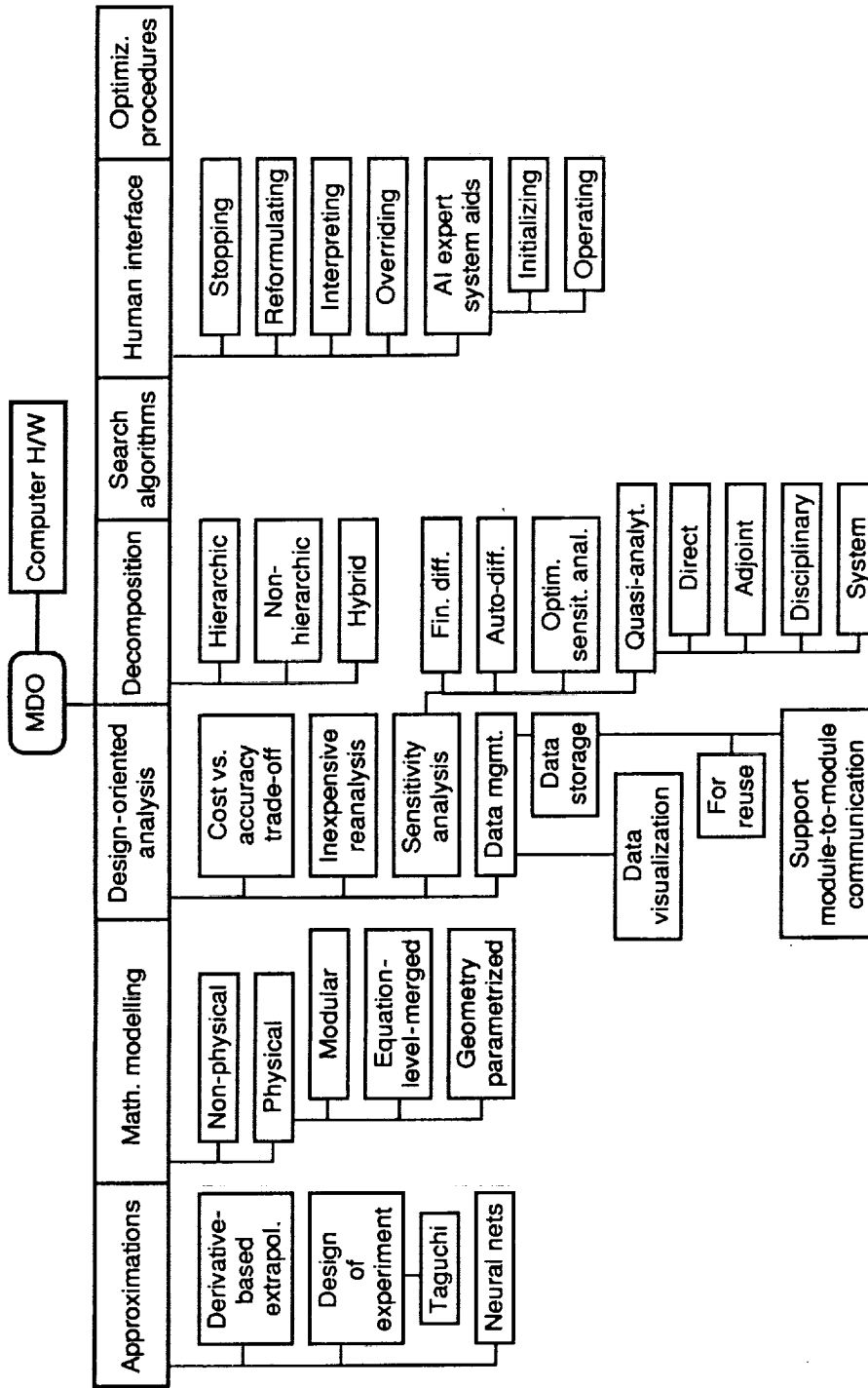


Figure 1. MDO principal conceptual components and their breakdown.

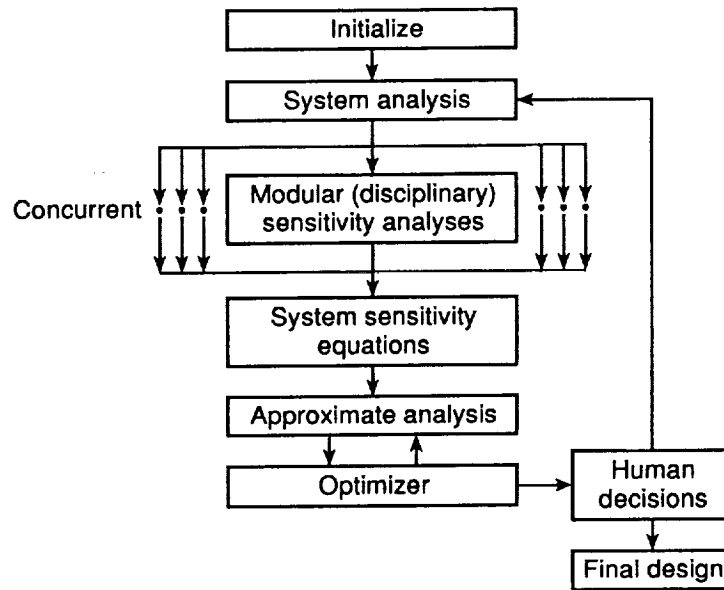


Figure 2. An optimization procedure for a non-hierarchical system (note an opportunity for concurrent processing at the level of disciplinary sensitivity analyses).

1611

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE May 1993	3. REPORT TYPE AND DATES COVERED Technical Memorandum
----------------------------------	----------------------------	--

4. TITLE AND SUBTITLE Multidisciplinary Design Optimization: An Emerging New Engineering Discipline	5. FUNDING NUMBERS 505-63-50
6. AUTHOR(S) Jaroslaw Sobieszczanski-Sobieski	

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) NASA Langley Research Center Hampton, VA 23681-0001	8. PERFORMING ORGANIZATION REPORT NUMBER
--	--

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington DC 20546-0001	10. SPONSORING / MONITORING AGENCY REPORT NUMBER NASA TM-107761
--	--

11. SUPPLEMENTARY NOTES
To be presented at the World Congress on Optimal Design of Structural Systems, Rio de Janeiro, Brazil, August 2-6, 1993.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Unclassified - Unlimited Subject Category 05	12b. DISTRIBUTION CODE
---	------------------------

13. ABSTRACT (Maximum 200 words)

This paper defines the Multidisciplinary Design Optimization (MDO) as a new field of research endeavor and as an aid in the design of engineering systems. It examines the MDO conceptual components in relation to each other and defines their functions.

14. SUBJECT TERMS Multidisciplinary, system, analysis, optimization, design	15. NUMBER OF PAGES 12
	16. PRICE CODE A03

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT Unlimited
---	--	---	---