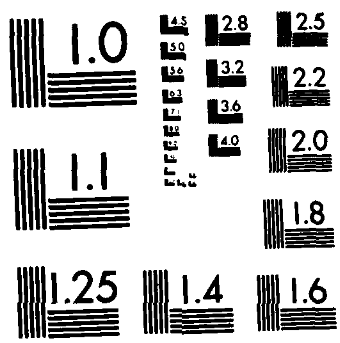
AD-A128 609 A COMPARATIVE STUDY OF OPTIMIZATION ALGORITHMS FOR ENGINEERING SYNTHESIS(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA C M SPRAGUE MAR 83 1/2

UNCLASSIFIED F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A COMPARATIVE STUDY OF OPTIMIZATION
ALGORITHMS FOR ENGINEERING SYNTHESIS

by

Chester Michael Sprague

March 1983

Thesis Advisor: G. Vanderplaats

Approved for public release; distribution unlimited.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A128 609	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Comparative Study of Optimization Algorithms for Engineering Synthesis		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; March 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Chester Michael Sprague		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 1983
		13. NUMBER OF PAGES 113
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Comparative Studies
Optimization
Engineering Synthesis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A variety of optimization algorithms for engineering synthesis are included in a new general-purpose optimization computer program called ADS-1 (Automated Design Synthesis, Version 1). Preliminary testing of all presently available algorithms is conducted utilizing several carefully selected problems of significant size and complexity. These include a problem with 56 design variables and over 3500 inequality constraints.

#20 - ABSTRACT - (CONTINUED)

The capabilities and utility of the ADS program coupled with a structural analysis code utilizing finite element techniques is demonstrated and numerical results are presented that compare the relative efficiency and reliability of the various optimization algorithms. The number of function and gradient calculations are considered important measures of merit in comparing the various algorithms.

A comparison of results with another existing optimization computer code is included to document the accuracy and reliability of the ADS program. Preliminary testing of the ADS program demonstrates the flexibility a design engineer would have in selecting an optimization algorithm best suited to solve a particular problem.

Approved for public release; distribution unlimited.

A Comparative Study of Optimization Algorithms for Engineering Synthesis

by

Chester Michael Sprague
Lieutenant Commander, United States Coast Guard
B.S., United States Coast Guard Academy, 1969

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

March 1983

Author: Chester M. Sprague

Approved by: Garret N. Vanderplaats
Thesis Advisor

David Salinas
Second Reader

P. J. Marto
Chairman, Department of Mechanical Engineering

J. M. Dyer
Dean of Science and Engineering

ABSTRACT

A variety of optimization algorithms for engineering synthesis are included in a new general-purpose optimization computer program called ADS-1 (Automated Design Synthesis, Version 1). Preliminary testing of all presently available algorithms is conducted utilizing several carefully selected problems of significant size and complexity. These include a problem with 56 design variables and over 3500 inequality constraints.

The capabilities and utility of the ADS program coupled with a structural analysis code utilizing finite element techniques is demonstrated and numerical results are presented that compare the relative efficiency and reliability of the various optimization algorithms. The number of function and gradient calculations are considered important measures of merit in comparing the various algorithms.

A comparison of results with another existing optimization computer code is included to document the accuracy and reliability of the ADS program. Preliminary testing of the ADS program demonstrates the flexibility a design engineer would have in selecting an optimization algorithm best suited to solve a particular problem.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to express my gratitude to Professor Vanderplaats for his patience and support as well as for the many long hours he spent helping me, no problem was too trivial despite the fact that he was engaged in many important projects himself. Professor Vanderplaats would also take the time to help me and other students in the curriculum develop and mature as engineers; for this behind-the-scenes demonstration of engineering leadership I wish to express my admiration.

Dr. Hiroyuki Sugimoto was an unexpected source of invaluable help and advice. I would like to thank him for the many hours he devoted in helping me accomplish this research.

Words cannot adequately express my appreciation to my wife, Penny for her support, and encouragement. She willingly accepted more than her share of the responsibility of raising three children during this protracted period of time. My children, Wendy, Tania, and Joe also deserve all the praise I can give them for their understanding and tolerance of my many hours away from home.

This research would not have been possible without the guidance of Professor Vanderplaats and the patient understanding of my family; thank you all very much.

I. INTRODUCTION

A. BACKGROUND

The concept of structural synthesis, a new general approach to structural optimization, was popularized by Schmit in 1960 [Ref. 1]. Structural synthesis, simply stated, couples finite element structural analysis with non-linear mathematical programming techniques. Schmit reasoned that the design of structures for minimum weight was, after all, simply the classic problem of allocation of scarce resources. He emphasized the importance of considering a multiplicity of distinct loading conditions and the need for inequality constraints to deal with a variety of different failure modes simultaneously, as well as *side constraints* (or bounds) on the size of the elements in the structure [Ref. 2].

Numerical techniques to solve the general non-linear, inequality constrained optimization problem developed rapidly after 1960. It was the advances of the high speed digital computer however, that allowed the science to fully mature. In fact the state of the art in mathematical programming is such, that the design engineer today should not find it necessary to develop his own computer program considering the widely available existing codes and the prohibitive costs of developing a new optimization code. The state of the art in finite element analysis has also enjoyed a considerable advancement. Thus there exists today the ability to efficiently design complex structures with many design variables under multiple loading conditions subject to a variety of constraints including stress, displacement, buckling and frequency as examples.

Structural synthesis continues to be the subject of active research; two specific areas for further study have been identified by Vanderplaats in [Ref. 3]. First is the need for public availability of a computer code incorporating a variety of optimization algorithms that reflect the state of the art in optimization. Secondly, the efficiency, reliability and accuracy of the various algorithms need to be compared and the results well documented. With this information, the engineer who may not have written his own optimization code, would be able to intelligently select the appropriate algorithm with only a basic knowledge of structural synthesis concepts, and tailor the algorithm to suit a particular problem.

The ADS library of design optimization algorithms was developed by Vanderplaats in response to the first need for a new general-purpose optimization computer code [Ref. 4]. ADS is unique insofar as it incorporates in a single program, a variety of different optimization algorithms. The purpose of this research is to perform some of the preliminary testing of this code and document the comparative studies. The specific objectives of this thesis as well as the details of the development of the various computer codes will be discussed in the remaining sections of this chapter.

B. THESIS OBJECTIVES

The primary objective of this thesis is to conduct the preliminary testing of all presently available algorithms in the ADS program. Although more than two years in development, numerous programming bugs remain to be ferreted out. Furthermore, some algorithms had never been tested with problems of significant size. Various default values for control parameters will also be determined by the preliminary testing.

While testing is in progress a second primary objective is to compare and document the efficiency of programming, reliability of results and accuracy of solutions of the various algorithms. To insure validity of the comparative study all testing is to be accomplished in accordance with the following requirements:

1. The same person is to test all algorithms on the same computer. The mainframe computer used in this research is an IBM 3033 system 370.
2. Default values will be used in the comparative studies. "Fine tuning" of algorithms by overriding default settings will be avoided insofar as possible.
3. Test cases of significant size and complexity will be selected for their potential to demonstrate the utility and flexibility of the ADS program and not because of their known ability to work well on a given algorithm.

Finally, a secondary objective is to compare results to the solutions provided by CONMIN [Ref. 5], a Fortran program for constrained minimization developed by Vanderplaats in 1973. CONMIN is considered well tested and reliable; the comparison thus rendered should lend credence to the results.

C. DEVELOPMENT OF COMPUTER PROGRAMS

1. ADS-1 (Automated Design Synthesis, Version 1)

The primary motivation behind ADS-1 was the need to provide a selection of optimization algorithms in a sophisticated computer code that could be applied to a variety of design problems. The ability to easily override default values of control parameters further enhances the flexibility of the program to be tailored to suit the particular design problem at hand.

The ADS program [Ref. 4] is written in subroutine form, well documented internally, and contains pseudo-dynamic dimensioning to maximize the efficient use of storage in the computer. Due to its inherent modularity the program is easy to interrupt and restart and amenable to multi-level optimization. These features add to its portability and reflect the state of the art in modern programming practices.

COPES, the control program for invoking CONMIN [Ref. 6], was modified for use with ADS and is named "COPESA", whereby data transfer into and out of ADS is readily accomplished.

The solution of an optimization problem is divided into three user defined levels:

1. STRATEGY--The method of optimization used may be direct, where control is transferred directly to the optimizer, or indirect as in various penalty function methods. A complete list of strategies is in Table I.
2. OPTIMIZER--Algorithms presently include methods for unconstrained functions as well as direct methods for constrained methods. A complete list of optimizers is in Table II.
3. ONE-DIMENSIONAL SEARCH--The user is given a choice of curve fitting a polynomial with or without finding bounds, using the Golden Section method or using a combination of polynomial and Golden Section methods. A complete list of one-dimensional search techniques is in Table III.

The program assumes the user is knowledgeable enough to select an appropriate combination of strategy, optimizer and one-dimensional search. For example, it would not be appropriate to use a variable metric optimizer on a constrained optimization problem unless one of the penalty function strategies was specified. Table IV lists the available options and feasible combinations are indicated.

2. <u>SADT (Structural Analysis and Design--Trusses)</u>

The primary purpose of SADT by Fitzgerald in [Ref. 36] was the development of a finite element code for three-dimensional indeterminate truss analysis and design. The code was written such that it could be easily coupled to an

TABLE I

Strategy Options in ADS

ISTRAT	STRATEGY TO BE USED
0	None. Go directly to the optimizer.
1	Sequential unconstrained minimization using the quadratic exterior penalty function method [Refs. 7 and 8].
2	Sequential unconstrained minimization using the linear extended interior penalty function method [Refs. 9 through 11].
3	Sequential unconstrained minimization using the quadratic extended interior penalty function method [Ref. 12].
4	Sequential unconstrained minimization using the cubic extended interior penalty function method [Refs. 13 and 14].
5	Augmented Lagrange multiplier method [Refs. 15 through 19].
6*	Sequential Linear Programming [Refs. 20 and 21].
7*	Method of Centers (Method of Inscribed Hyperspheres) [Ref. 22].
8*	Powell's Variable Metric Method for Constrained Minimization [Refs. 17, 23 and 24].

*Not available as of February, 1983

TABLE II

Optimizer Options in ADS

IOPT	OPTIMIZER TO BE USED
0	None. Go directly to one-dimensional search. This option should be used only for program development.
1	Method of Feasible Directions (MFD) for constrained minimization [Refs. 25 and 26].
2	Fletcher-Reeves algorithm for unconstrained minimization [Ref. 27].
3	Robust Method of Feasible Directions for constrained minimization [Ref. 28].
4	Davidon-Fletcher-Powell (DFP) variable metric method for unconstrained minimization [Refs. 29 and 30].
5	Broydon-Fletcher-Goldfarb-Shanno (BFGS) variable metric method for unconstrained minimization [Refs. 31 through 34].
6*	Random Search for unconstrained minimization.
7*	Random Search for constrained minimization.
8*	Newton's Method for unconstrained minimization.
9*	Quadratic Programming [Ref. 35].

*Not available as of February, 1983

TABLE III

One-Dimensional Search Options in ADS

IONED	ONE-DIMENSIONAL SEARCH OPTION [Refs. 7 and 52]
1	Find brackets on the minimum of an unconstrained function.
2	Find the minimum of an unconstrained function using the Golden Section method.
3	Find the minimum of an unconstrained function using the Golden Section method, followed by cubic polynomial interpolation.
4	Find the minimum of an unconstrained function by first finding bounds and then using polynomial interpolation.
5	Find the minimum of an unconstrained function by polynomial interpolation/extrapolation without first finding bounds on the solution.
6	Find brackets on the minimum of a constrained function.
7	Find the minimum of a constrained function using the Golden Section method.
8	Find the minimum of a constrained function using the Golden Section method, followed by cubic polynomial interpolation.
9	Find the minimum of a constrained function by first finding bounds and then using polynomial interpolation.
10	Find the minimum of a constrained function by polynomial interpolation/extrapolation without first finding bounds on the solution.

TABLE IV

Program Options in ADS

	OPTIMIZER									
STRATEGY	0	1	2	3	4	5	6*	7*	8*	9*
0	X	X	X	X	X	X	X	X	X	X
1	0	0	X	0	X	X	X	0	X	0
2	0	0	X	0	X	X	X	0	X	0
3	0	0	X	0	X	X	X	0	X	0
4	0	0	X	0	X	X	X	0	X	0
5	0	0	X	0	X	X	X	0	X	0
6*	0	X	0	X	0	0	0	0	0	0
7*	0	X	0	X	0	0	0	0	0	0
8*	0	X	0	X	0	0	0	0	0	X
ONE-D SEARCH										
1	X	0	0	0	0	0	X	0	0	0
2	X	0	X	0	X	X	X	0	X	X
3	X	0	X	0	X	X	X	0	X	X
4	X	0	X	0	X	X	X	0	X	X
5	X	0	X	0	X	X	0	X	0	0
6	X	0	0	X	0	0	0	X	0	0
7	X	X	0	X	0	0	0	X	0	0
8	X	X	0	X	0	0	0	X	0	0
9	X	X	0	X	0	0	0	X	0	0
10	X	X	0	X	0	0	0	X	0	0

X = Allowed Combination

0 = Combination Not Allowed

* = Not Available as of February, 1983

optimizer for comparative studies. A secondary objective was to provide a user-friendly computer code that could be employed for truss analysis only. SADT was therefore selected as the analysis code for test cases involving trusses and space towers.

Design variables may include member element cross sectional areas, nodal coordinates, or both. A well written user's manual is included in [Ref. 36] and provides necessary details for coupling the program to an optimizer as well as for test case data preparation.

The finite element method of analysis is used for static analysis, and eigenvalues are computed according to the subspace iteration technique when frequency constraints are specified [Ref. 37]. Multiple static loading conditions can be accommodated as well as constraints on stress, Euler buckling, displacement and the first fundamental frequency of the structure. The objective function is minimum weight of the structure. Side constraints may be imposed on the upper and/or lower bounds of the design variables. Design variable linking is permitted for both member areas and coordinates. The user may specify different materials for the various members. All loads are assumed concentrated at the joints and the truss is treated as a discrete, pin-connected structure.

D. PREVIOUS COMPARATIVE STUDIES

Even though many methods are available for solving the constrained, non-linear optimization problem there has been relatively little research done in the way of comparative studies since the inception of structural synthesis in 1960.

Colville, in a landmark study in 1968, sent eight constrained problems (three to 16 design variables each) to the developers of 30 different codes. Solution times as well as preparation time and the number of function and constraint evaluations were requested from each participant [Ref. 38]. Colville placed great emphasis on solution times and therefore developed a standard timing routine in an attempt to normalize solution times to eliminate differences among computers. He could not of course, eliminate the differences in the developers' abilities to efficiently code their problems for solution.

Eason and Fenton tested 13 different problems on 20 different codes in 1972 [Ref. 39]. They effectively eliminated the problems evident in Colville's study. All of their test case problems however, had fewer than seven independent design variables.

Sangren and Ragsdell conducted a comparative study on 30 problems in [Ref. 40]. The number of design variables in this study range from two to 48 while the number of constraints range from zero to 19.

The problems selected for comparative study in this research have from 5 to 56 design variables and from 11 to

3550 constraints, the largest problem being the design of a 234-bar space tower subject to constraints on stress, Euler buckling, and displacement of joints.

II. OPTIMIZATION TECHNIQUES

A. OPTIMIZATION CONCEPTS

The general, non-linear, constrained optimization problem can be stated mathematically as:

Minimize:

$$F(\underset{\sim}{X}) \qquad (2.1)$$

Subject to:

$$G_j(\underset{\sim}{X}) \leq 0 \qquad j = 1,NCON \qquad (2.2)$$

$$X_i^{\ell} \leq X_i \leq X_i^{u} \qquad i = 1,NDV \qquad (2.3)$$

$F(\underset{\sim}{X})$ is called the objective function. It is the function with respect to which the design is optimized. It may be a linear or non-linear function of the design variables $\underset{\sim}{X}$. Generally speaking, the objective function may be implicit or explicit functions of $\underset{\sim}{X}$. It is important however, that these functions be continuous and have continuous first derivatives in $\underset{\sim}{X}$. The $G_j(X)$ inequalities define the constraints which the user imposes on the design. Equation 2.3 defines side constraints or bounds on the design and are the limits over which $F(\underset{\sim}{X})$ and $G(\underset{\sim}{X})$ are defined. If the inequality condition of equation 2.2 is not met for any constraint, that constraint is said to be violated. If the equality condition of equation 2.2 is met then the constraint is called active.

The ability to deal with equality constraints is also included in the ADS program. This feature was not fully operational at the time of this writing however, and therefore was not tested.

The n-dimensional space spanned by the design variables $\tilde{X}$ is referred to as the design space. Any design satisfying equations 2.2 and 2.3 is a feasible design and the minimum feasible design is said to be optimal. Problems in optimization may be classified according to whether or not they are constrained. Algorithms to solve these problems are therefore generally classified by the type of problem they were developed to solve efficiently. In the remaining sections of this chapter the algorithms used in the preliminary testing of the ADS library will be discussed. Techniques to solve the unconstrained minimization problem will be discussed first, followed by constrained minimization methods. Lastly, the various techniques for minimizing functions of one variable, the so-called one-dimensional search, will be discussed. These techniques are called upon by both major categories of algorithms to solve a sub-problem in the optimization task, wherein the following recursive relationship is commonly employed:

$$\tilde{X}^q = \tilde{X}^{q-1} + \alpha^* \tilde{S}^q \qquad (2.4)$$

in this equation q is the iteration number, α^* is the scalar step size and $\underset{\sim}{S}$ is the vector search direction.

B. UNCONSTRAINED MINIMIZATION

1. Introduction

In the general case of unconstrained minimization of a multi-variable function, the calculus requires for a minimum solution, that the gradient of the objective function with respect to the design variables equate to zero and that the Hessian matrix of second partial derivatives of the objective function with respect to the design variables be positive definite (all eigenvalues > 0). If the Hessian matrix is positive definite a relative minimum at least is guaranteed. Unconstrained methods are therefore, intrinsically concerned with gradient information; as a result, they are classified according to the type of derivative information they require. Zero-order methods such as Random Search and Powell's Conjugate Directions Method are non-gradient methods whereas first-order methods such as Fletcher-Reeves require first derivative information only and so on. These methods as well as the variable metric methods of Davidon-Fletcher-Powell and Broydon-Fletcher-Goldfarb-Shanno will be discussed in the next few sections.

2. Non-Gradient Methods

a. Random Search

Random Search methods represent the simplest possible approach to optimization, wherein a randomly selected large number of possible X vectors are evaluated for values of the objective functions. The X vector corresponding to the least objective function is the optimal design. There are many drawbacks, not the least of which is efficiency. The necessity to evaluate a large number of possible designs is required to insure a precise optimum has been obtained. The need to improve efficiency is the motivation behind many of the modifications available for random search methods. These methods lend themselves well to coding on a hand-held calculator, furthermore they require little storage on the computer, making them efficient from that point of view.

b. Powell's Conjugate Directions Method

Powell's method is certainly the most popular, if not the most efficient, of all zero-order methods. Powell's Method is based on the concept of conjugate directions. The algorithm requires an initial search in n-orthogonal directions wherein each search updates the X vector according to equation 2.4.

The new search direction is found by simply connecting the first and last design points; this becomes the n+1 conjugate search direction. Powell's Method breaks down if a search direction makes no improvement because subsequent search directions will not be conjugate. A second well recognized problem is the tendency after a few iterations for the search directions to become nearly parallel. Powell offers a sophisticated technique to overcome this second problem [Ref. 41]. Simply restarting the process with unidirectional searches is an effective, if not elegant, way of dealing with this problem as noted in [Ref. 42]. Powell's Method is not presently available in ADS.

The next logical step in sophistication is to provide gradient information to the optimizer. In the following sections the Fletcher-Reeves algorithm and variable metric methods will be discussed insofar as they are first-order methods presently available in ADS.

3. Gradient Methods

a. Fletcher-Reeves Method of Conjugate Directions

The Fletcher-Reeves algorithm is actually a modification of the steepest descent algorithm with a significant improvement in the rate of convergence. The basic approach is to pick conjugate search directions according to:

$$\underset{\sim}{S}^{q} = -\underset{\sim}{\nabla}F(\underset{\sim}{X}^{q}) + \beta_q \underset{\sim}{S}^{q-1} \tag{2.5}$$

where:

$$\beta_q = |\underset{\sim}{\nabla}F(\underset{\sim}{X}^{q})|^2 / |\underset{\sim}{\nabla}F(\underset{\sim}{X})^{q-1}|^2 \tag{2.6}$$

The initial search direction is in the direction of steepest descent:

$$\underset{\sim}{S}^{q} = -\underset{\sim}{\nabla}F(\underset{\sim}{X}^{q}) \tag{2.7}$$

The method is conceptually similar to Powell's Method, except now each search direction is conjugate. Theoretically, convergence for a quadratic function in n or fewer iterations can be expected, however, restarting the process every few iterations as in Powell's Method is usually required.

b. Variable Metric Methods

Variable Metric Methods retain information about previous iterations also. In these methods a matrix $\underset{\approx}{H}$ is created which approximates the inverse of the Hessian matrix. The search direction is defined at iteration q as follows:

$$\underset{\sim}{S}^q = -\underset{\approx}{H}\underset{\sim}{\nabla}F(\underset{\sim}{X}^q) \tag{2.8}$$

Again the initial search direction is determined by the method of steepest descent. At the end of iteration q, the $\underset{\approx}{H}$ matrix is updated according to:

$$\underset{\approx}{H}^{q+1} = \underset{\approx}{H}^q + \underset{\approx}{D}^q \tag{2.9}$$

where $\underset{\approx}{D}^q$ is a symmetric matrix determined according to the following formulation:

$$\underset{\approx}{D}^q = [\sigma + \theta\tau/\sigma^2]\underset{\sim}{p}\underset{\sim}{p}^T \tag{2.10}$$

the terms in this equation are defined as:

$$\underset{\sim}{p} = \underset{\sim}{X}^q - \underset{\sim}{X}^{q-1} \qquad (2.11)$$

$$\underset{\sim}{y} = \nabla F(X^q) - \nabla F(X^{q-1}) \qquad (2.12)$$

$$\sigma = \underset{\sim}{p} \cdot \underset{\sim}{y} \qquad (2.13)$$

$$\tau = \underset{\sim}{y}^T \, \underset{\approx}{H}^q \, \underset{\sim}{y} \qquad (2.14)$$

and θ is a parameter used to select the form of the update formula, equation 2.10. The Davidon-Fletcher-Powell Method sets $\theta = 0$ in equation 2.10 whereas the Broydon-Fletcher-Goldfarb-Shanno Method sets $\theta = 1$ [Ref. 42]. There are other possible algorithms in the class of variable metric methods but these two methods are the most popular and are presently available in ADS.

C. CONSTRAINED MINIMIZATION

1. Introduction

Constrained methods of minimization were developed to deal with problems that have limitations placed on a set of functions of the design variables. These limitations may be side constraints which directly impose bounds on the design variables, or so-called behavior constraints which are functions of the design variables. Behavior constraints may take the form of equality or inequality constraints, but in either case the design must satisfy the behavior constraints while staying within the bounds imposed by the side constraints.

Direct methods consider the constraints as limiting hypersurfaces and attempt to directly minimize the objective function in their presence. In contrast, the so-called penalty function methods transform the constrained minimization problem into a sequence of unconstrained minimization problems. Although direct methods are often more efficient, indirect methods are popular because they are simple to invoke. The engineer must employ an appropriate unconstrained minimization algorithm when using a penalty function method.

The indirect methods utilizing penalty function techniques may be further classified into two broad categories: interior and exterior. Interior methods are designed to approach the optimum from the feasible region whereas the exterior methods approach the solution from the infeasible sector. A pseudo-objective function is created by imposing a penalty for violated constraints. The general technique is to minimize this pseudo-objective function as an unconstrained problem. The methods require repetitive solution to a series of unconstrained problems thus the term, "Sequential Unconstrained Minimization Techniques" (SUMT), is applied to this broad class of indirect methods.

2. Direct Methods

Most optimization algorithms proceed iteratively toward a solution from a user supplied initial $\underset{\sim}{X}$ vector which may or may not define a feasible design. The design is modified according to the recursive relationship:

$$\tilde{X}^{q+1} = \tilde{X}^{q} + \alpha^{*}\tilde{S} \qquad (2.15)$$

where q is the iteration number, $\tilde{S}$ is a vector search direction in the design space and the scalar, α^*, defines the distance the optimizer moves in the search direction $\tilde{S}$. The choice of $\tilde{S}$ is such that the objective function is reduced. The efficiency and reliability of a given optimization algorithm is largely due to the fundamental method of determination of the search direction $\tilde{S}$ and the step size α^*. These methods will be discussed in the next few sections of this chapter.

a. Method of Feasible Directions

Optimization in the Method of Feasible Directions proceeds in two basic steps, first a usable-feasible search direction is determined, then a one-dimensional search is performed in this direction to reduce the objective as much as possible without violating constraints. The method assumes that the initial X vector of design variables defines a feasible design. A usable-feasible search direction to improve this design is found by solving the following subproblem:

$$\text{Maximize:} \quad \beta \qquad (2.16)$$

Subject to:

$$\tilde{\nabla} F(\tilde{X}) \cdot \tilde{S} + \beta \leq 0 \qquad (2.17)$$

$$\underset{\sim}{\nabla} G_j(\underset{\sim}{X}) \cdot S + \theta_j \cdot \beta \leq 0 \qquad j \in J \tag{2.18}$$

$$\underset{\sim}{S} \cdot \underset{\sim}{S} \leq 1 \tag{2.19}$$

where J is the set of currently active constraints, $G_j(\underset{\sim}{X}) = 0$. $\underset{\sim}{\nabla}$ is the gradient operator and the components of θ are referred to as push-off factors, which act to push the design away from currently active constraints. A value of unity for θ will yield a search direction which approximately bisects the usable-feasible sector.

If the initial design is infeasible it is possible to find a search direction that will direct the design to the feasible region [Ref. 42].

Using equation 2.19 with equations 2.16 through 2.18 results in a linear problem of finding $\underset{\sim}{S}$ except for one quadratic constraint. Zoutendijk in [Ref. 25] provides a direct approach to overcome this difficulty. A detailed explanation of these techniques is provided in [Ref. 42].

The method then proceeds to update the design in accordance with equation 2.15. This step is commonly performed by polynomial interpolation but a variety of one-dimensional search methods may be used.

b. Robust Method of Feasible Directions

The Robust Method of Feasible Directions is a new algorithm presently being developed by Vanderplaats, and incorporates the best features of the Method of Feasible Directions (MFD) and the Generalized Reduced Gradient (GRG)

Method [Ref. 28]. Only gradients of active constraints are required in the MFD, which is considered an attractive feature, while the GRG method has the nice feature of precisely following the constraint boundaries from one vertex to the next without the need to move away from the constraints. The Robust MFD retains these desirable features but does not require the addition of slack variables peculiar to the GRG method, thus avoiding the large matrix operations associated with the GRG method. The method involves solving the following search direction sub-problem:

Maximize:

$$- \underset{\sim}{\nabla} F(\underset{\sim}{X}) \cdot S \tag{2.20}$$

Subject to:

$$\underset{\sim}{\nabla} G_j(\underset{\sim}{X}) \cdot \underset{\sim}{S} \leq 0 \qquad j \in J \tag{2.21}$$

$$\underset{\sim}{S} \cdot \underset{\sim}{S} < 1 \tag{2.22}$$

This is the same form as the direction finding sub-problem in MFD except the dimensionality is reduced by the elimination of the variable β. The following advantages in determining the search direction in this manner are repeated here from [Ref. 28] for convenience:

1. The dimensionality of the design problem is not increased by the addition of slack variables to the inequality constraints.

2. The algorithm for finding S is specifically designed for inequality constrained problems, thus improving efficiency.
3. Only gradients of active constraints are required.
4. The number of dependent variables is greatly reduced in comparison to the GRG method, thus a reduction in the size of the sub-problem in the one-dimensional search is achieved.

Equality constraints are effectively handled as a special case of inequality constraints. Initially infeasible designs require a modification to the search direction-finding sub-problem where the violated constraints are treated as inequality constraints. A direction to the feasible region is then determined in a manner similar to the Method of Feasible Directions.

The Robust method incorporates a particularly attractive feature of infrequent gradient calculations. That is, gradients of active constraints are treated as constants for several iterations thus greatly reducing the computational cost of the algorithm. It should be noted that if infrequent gradient calculations are not used the method yields the same results as the GRG Method.

The one-dimensional search is performed in the same manner as for the GRG method. Significant in this procedure is the fact that Newton's Method is employed to drive the active constraints corresponding to the dependent variables to zero. This procedure usually requires several iterations.

The Robust Method of Feasible Directions shares some of the limitations of the GRG method [Ref. 28].

1. It produces infeasible designs and relies on Newton's Method to return to the feasible region.
2. It has difficulty dealing with highly non-linear functions.
3. If the analysis is itself iterative the method may be unable to satisfy constraints due to the resulting instability.

3. Indirect Methods

ADS incorporates several SUMT methods, namely, exterior, extended interior, and Augmented Lagrange Multiplier (ALM) penalty function methods. The numerical ill-conditioning often encountered in SUMT methods is reduced in the ALM method. This method has therefore received wide attention in the literature and is included in the ADS library.

All SUMT methods create a pseudo-objective function of the general form:

$$\Phi(\underset{\sim}{X}, r_p) = F(\underset{\sim}{X}) + r_p P(\underset{\sim}{X}) \qquad (2.23)$$

where $F(\underset{\sim}{X})$ is the original objective function, $P(\underset{\sim}{X})$ is the penalty function and the multiplier, r_p, determines the magnitude of the penalty applied. The following sections discuss in more detail the technique of determining $P(X)$ which is the fundamental basis of each method.

a. Exterior Penalty Function Method

The basic mathematical formula for determining the penalty function $P(X)$ is:

$$P(\underset{\sim}{X}) = \sum_{j=1}^{m} \{MAX[0,q_j(\underset{\sim}{X})]\}^2 + \sum_{k=1}^{\ell} [h_k(\underset{\sim}{X})]^2 \qquad (2.24)$$

A penalty is imposed if, and only if, an inequality, $G_j(\underset{\sim}{X})$, or equality, $H_k(\underset{\sim}{X})$, constraint is violated. The "offending" constraint is squared to provide a slope of zero for the penalty function at the constraint boundary thus insuring a continuous first derivative for the pseudo-objective function. The second derivative is not required to be continuous however, therefore if second-order methods are employed in the unconstrained minimization, numerical ill-conditioning may result [Ref. 42].

The multiplier, r_p, is critical in this method as it is in all SUMT methods If r_p is chosen small the pseudo-objective function is easily minimized but may result in extreme constraint violation; whereas a large r_p will guard against this, the resulting problem is usually numerically ill-conditioned. Therefore the algorithm starts with a small r_p, which is then increased by a factor γ. At each iteration Φ is minimized starting from the previous optimum solution.

As r_p is increased in the sequential optimization process, the pseudo-objective function becomes increasingly non-linear. The constrained optimum solution is also approached from the infeasible region. In other words the optimum is approached with a series of infeasible designs, none of which are usable. The interior penalty function method approaches the optimum from the feasible sector with a series of improving feasible designs. This attractive feature is discussed in the next section.

b. Interior Penalty Function Method

The most common formulation for the penalty function in this method is:

$$P(X) = \sum_{j=1}^{m} [-1./g_j(\underset{\sim}{X})] \tag{2.25}$$

resulting in a more complicated pseudo-objective function to minimize:

$$\Phi(\underset{\sim}{X}, r_p', r_p) = F(\underset{\sim}{X}) + r_p'(P(\underset{\sim}{X})] + r_p \sum_{k=1}^{\ell} [h_k(\underset{\sim}{X})]^2 \tag{2.26}$$

Note that equality constraints (h_k) are dealt with in the same manner by interior and exterior methods. The significant difference between the methods, besides the formulation of $P(\underset{\sim}{X})$, is the fact that in interior methods the penalty parameter, r_p', is sequentially decreased with every SUMT iteration, while in exterior methods r_p is sequentially increased. Interior methods result in the approach of the optimum solution from the feasible region as $r_p \to 0$, but is discontinuous at constraint boundaries. The exterior method, on the other hand, is well-defined everywhere, but leads to an optimum solution only in the limit as $r_p \to \infty$. The extended interior penalty methods are designed to incorporate the best features of both methods by effecting a transition between the interior and exterior methods at a point in the optimization task. Needless to say, this transition point is critical and therefore of fundamental concern in the various extended interior penalty function methods, which are discussed next.

c. Extended Interior Penalty Function Method

The chief advantage of the interior penalty method is that it results in a sequence of improving feasible designs from an initially acceptable starting point. This desirable feature is maintained in this method by a judicious selection of the parameter, ε, in the formulation of the penalty function P(X):

$$P(\underset{\sim}{X}) = \sum_{j=1}^{m} g_j(\underset{\sim}{X}) \tag{2.27}$$

where

$$g_j(\underset{\sim}{X}) = -1./g_j(\underset{\sim}{X}) \quad \text{if} \quad g_j(\underset{\sim}{X}) \leq \varepsilon \tag{2.28}$$

$$g_j(\underset{\sim}{X}) = -[2. - g_j(\underset{\sim}{X})]/\varepsilon^2 \quad \text{if} \quad g_j(\underset{\sim}{X}) > \varepsilon \tag{2.29}$$

The parameter, ε, is a small negative number and signifies the transition from the interior to the exterior methods [Ref. 42]. These equations define the linear extended interior penalty function. Because the second derivative of $\Phi(\underset{\sim}{X}, r_p', r_p)$ is discontinuous, Haftka and Starnes created the quadratic extended interior penalty function by changing equation 2.29 to:

$$g_j(\underset{\sim}{X}) = -1./\varepsilon\{[g_j(\underset{\sim}{X})/\varepsilon]^2 - 3.[g_j(\underset{\sim}{X})/\varepsilon] + 3.\} \tag{2.30}$$

$$\text{if} \quad g_j(\underset{\sim}{X}) > \varepsilon$$

Again the degree of non-linearity of Φ is increased as a price for the second-order continuity.

The linear and quadratic extended interior penalty methods are both critically dependent on the selection of ε. Haftka and Starnes recommend that ε be determined according to:

$$\varepsilon = -C(r_p')^a \qquad 1/3 < a < 1/2 \tag{2.31}$$

where C is a constant. At the beginning ε is chosen in the range $-.3 < \varepsilon < -.1$ and r_p' is chosen such that the objective and pseudo-objective functions are equal; the resultant value of C is thus determined [Ref. 43].

The quadratic extended interior penalty method has the disadvantage that the penalty increases dramatically for badly violated constraints. The variable penalty function method attempts to overcome this difficulty while continuing to insure second order continuity at the transition point. The selection of ε in the variable penalty method is recommended by Prasad in [Ref. 44] as follows:

$$\varepsilon = -\beta (r_p')^q \qquad (2.32)$$

where

$$1/(2+S) < q < 1/S \quad \text{for} \quad S > 0 \qquad (2.33)$$

and β is a positive constant chosen such that ε is initially near zero. In ADS, the variable penalty method is used wherein $S = 3$ thus the strategy is referred to as the cubic extended interior penalty function method.

d. Augmented Lagrange Multiplier Method

The efficiency of SUMT methods can be improved by the inclusion of Lagrange multipliers, thus reducing dependency of the algorithm on the choice of the penalty parameters. The Lagrangian is created for equality constrained problems as follows:

$$L(\underset{\sim}{X},\underset{\sim}{\lambda}) = F(\underset{\sim}{X}) + \sum_{k=1}^{\ell} \lambda_k h_k(\underset{\sim}{X}) \qquad (2.34)$$

Since the minimum of the Lagrangian provides the solution to the general equality constrained problem, a pseudo-objective function, called the augmented Lagrangian is created using the exterior penalty function method:

$$A(\underset{\sim}{X},\underset{\sim}{\lambda},r_p) = F(\underset{\sim}{X}) + \sum_{k=1}^{\ell}\{\lambda_k h_k(\underset{\sim}{X}) + r_p[h_k(\underset{\sim}{X})]^2\} \tag{2.35}$$

The method starts with the following values for λ.

$$\lambda_k = +1. \quad \text{if} \quad \underset{\sim}{\nabla} h_k(\underset{\sim}{X}) \cdot \underset{\sim}{\nabla} F(\underset{\sim}{X}) < 0 \tag{2.36}$$

$$\lambda_k = -1. \quad \text{if} \quad \underset{\sim}{\nabla} h_k(\underset{\sim}{X}) \cdot \underset{\sim}{\nabla} F(\underset{\sim}{X}) > 0 \tag{2.37}$$

The pseudo-objective function, $A(\underset{\sim}{X},\underset{\sim}{\lambda},r_p)$, is then minimized holding r_p and λ constant. A new set of Lagrange multipliers is calculated according to:

$$\lambda_k^{p+1} = \lambda_k^p + 2r_p h_k(\underset{\sim}{X}^p) \quad k = 1,\ell \tag{2.38}$$

The parameter r_p is sequentially increased as in the exterior SUMT method and the unconstrained minimization problem is solved for r_p and λ. The process is repeated until convergence is achieved.

The method is easily extended to handle inequality constraints by converting them to equivalent equality constraints by the addition of slack variables. A more complicated augmented Lagrangian is then formulated as the pseudo-objective function:

$$A(\underset{\sim}{X},\underset{\sim}{\lambda},\underset{\sim}{Z},r_p) = F(\underset{\sim}{X}) + \sum_{j=1}^{m}\{\lambda_j(q_j(\underset{\sim}{X})+Z_j^2) + r_p[g_j(\underset{\sim}{X}) + Z_j^2]^2\} \tag{2.39}$$

where there are m slack variables, Z_j^2. These are calculated as a sub-problem and so do not increase the dimensionality of the optimization task. Note that the pseudo-objective function has continuous first derivatives with respect to $\underset{\sim}{X}$ but discontinuous second derivatives at $g_j(\underset{\sim}{X}) = -\lambda_j/2r_p$; thus second order techniques should be avoided in the unconstrained minimization problem. The method has several attractive features repeated here from [Ref. 42].

1. The method is relatively insensitive to r_p, accordingly it is not necessary to increase r_p to ∞.
2. Equality constraints and inequality constraints precisely equal to zero are possible.
3. Acceleration to an optimum is achieved by updating the Lagrange multipliers.
4. The starting point may be feasible or infeasible.
5. At the optimum any Lagrange multiplier not equal to zero will identify an active constraint.

4. Other Constrained Minimization Methods

The discussion of optimization algorithms has been restricted to non-linear programming techniques insofar as these methods are fully operational in ADS. Sequential Linear Programming is another category of optimization techniques which will be included in the ADS library where a particular problem is linearized and a solution sought for the resulting linear approximation. Considering these techniques are, in theory, well-developed and quite effective this additional capability will enhance the utility of ADS.

The basic approach is to linearize the objective and constraint functions and obtain a solution to this approximation using the algorithm developed for linear programming. The process is iterative and therefore the techniques are referred to as Sequential Linear Programming (SLP). It is pointed out in [Ref. 42] that fully constrained problems usually converge rapidly while under-constrained problems often have difficulty in converging to an optimum solution. The difficulty may be overcome somewhat by sequential reduction of move limits on the optimizer. SLP characteristically produces a sequence of improving infeasible designs. The Method of Centers also called Method of Inscribed Hyperspheres, has the dual advantage of approaching the optimum with a sequence of improving feasible designs while following a path down the "center" of the design space. This method is discussed in the remainder of this section.

The basic approach in the Method of Centers is to inscribe a hypersphere in n-dimensional design space created when all of the constraints and objective function are linearized. The design then moves to the center of the hypersphere. This procedure is repeated to convergence within some user-specified tolerance. In the case of under-constrained problems the method is subject to the same problem as SLP in imposing move limits on the optimizer.

D. FUNCTIONS OF ONE VARIABLE: THE ONE-DIMENSIONAL SEARCH

1. Introduction

The one-dimensional search, as it is commonly referred to in algorithms for optimization, usually applies to determining α^*, the step size to be taken in the search direction $\underset{\sim}{S}$. Finding the minimum of any function of one variable is simply finding the point at which the first derivative vanishes. Since the function is not always an easily obtained analytic function in optimization, it is necessary to make some fundamental assumptions so that appropriate numerical analysis techniques may be brought to bear. Accordingly, the functions are assumed unimodal, that is, the function has only one relative minima in the region of concern. The functions are also assumed continuous as are their first and second derivatives. These assumptions will assure convergence to a minimum.

In the remaining sections of this chapter the methods used to conduct the one-dimensional search will be discussed.

2. Polynomial Approximation

The basic procedure in the polynomial approximation method is to evaluate the function at several points and then fit a polynomial curve to the data points using an appropriate curve-fitting technique. The minimum of this curve is approximately equal to the minimum of the true function. The method is simple, requires only a few function evaluations and is generally reliable for functions which are not too highly non-linear.

It is well known that a higher order polynomial will fit the data points more accurately; this gain in accuracy however can complicate the process of finding the minimum of the resulting polynomial. Also, interpolation between points is preferred to extrapolation beyond the region enclosed by the data points. The process of finding the minimum of the polynomial requires finding the point where the first derivative vanishes. Alternatively, there are numerical analysis techniques available to find the minimum or zeros of a higher order polynomial. These methods are not discussed here.

3. Golden Section Method

The Golden Section Method is popular because the rate of convergence is known and the requirements for function unimodality and continuity are relaxes. The disadvantage of the method lies in the inherently large number of function evaluations required as compared to other one-dimensional search methods.

The method involves picking two intermediate points, X_1 and X_2, between given upper and lower bounds, X_ℓ and X_u, such that $X_1 < X_2$. The function is then evaluated at X_1 and X_2 and one of the previous bounds is replaced by one of the intermediate points as follows:

$$F(X_1) > F(X_2) \qquad X_\ell \leftarrow X_1 \tag{2.40}$$

$$F(X_2) > F(X_1) \qquad X_u \leftarrow X_2 \tag{2.41}$$

The process is repeated until some user specified tolerance is satisfied. Fundamental to the method is the selection of the intermediate points. The Golden Section number, 1.61803, is used for this purpose:

$$(X_2 - X_\ell)/(X_1 - X_\ell) = 1.61803 \tag{2.42}$$

The Golden Section provides the ideal sequence for dividing the interval such that the minimum number of function evaluations is required. The advantage of this method is guaranteed accuracy whereas the relatively large number of function values required is a distinct disadvantage.

A similar method, the Fibonacci Search, based on the series of Fibonacci numbers, traps the minimum in successively smaller intervals. The Fibonacci Search is occasionally more efficient than Golden Section but is far more complicated.

4. Finding Bounds on the Solution

This method is usually used to obtain brackets on the solution, then Golden Section or polynomial methods are called to complete the one-dimensional search.

The method begins with an assumed initial lower bound X_ℓ and a proposed upper bound X_u. These two points are then evaluated as $F(X_\ell)$ and $F(X_u)$. If $F(X_u) \geq F(X_\ell)$ then X_u is the true upper bound. Assuming the slope of the function at X_ℓ is negative at X_ℓ, the solution is complete. If $F(X_u) < F(X_\ell)$ then the following update formula is applied iteratively to achieve the desired bounds:

$$X_u^{new} = (1 + a) X_u^{old} - a\, X_\ell^{old} \tag{2.43}$$

$$X_\ell^{new} = X_u^{old} \tag{2.44}$$

where a = golden section number = 1.61803.

Note that if the last three values of this iterative procedure are retained along with the function values, the three required points by the Golden Section and Polynomial methods are already available.

Many algorithms (e.g., MFD) require the constrained minimum of F(X). Polynomial and Golden Section methods are also used in ADS for this purpose. Note that the X used here is actually α^* in equation 2.4.

III. PRELIMINARY TESTING OF ADS-1

A. INTRODUCTION

Selection of test problems in a comparative study is of primary importance. Considering one of the objectives of this thesis is to demonstrate the utility and flexibility of the ADS library, test cases were selected from two fundamentally different areas in which optimization is commonly used. These areas are structural design (trusses, frames, space towers, etc.), and ship synthesis. There are many other areas in engineering where optimization is employed but the areas chosen here are selected for comparative study in this research due to the availability of the analysis codes.

A good test case is one in which no single constraint dominates the design. Three different truss cases were selected that met this criteria. They were also chosen because they are significant in size and complexity and thus would demonstrate the comparative efficiency and reliability of the various algorithms to be tested. Truss cases are popular in the literature because differences due to modeling details and idealizations can be eliminated easily; they also lend themselves well to finite element methods of analysis. Because analytically optimum solutions to the test cases are not available, solutions obtained by the well developed and thoroughly tested optimization program "CONMIN" are provided as a base-line for the results from ADS.

The remaining two test problems consist of a 10-variable cantilever beam optimized for minimum volume, and the conceptual design of the FFG-7 Perry Class Frigate where the objective function is taken to be the full load displacement of the vessel. The details of the various test cases are presented in the following sections.

B. DESCRIPTION OF TEST PROBLEMS

1. 10-Variable Cantilevered Beam

The 10-variable cantilever beam test case was developed by Vanderplaats in 1979 as a teaching aid for a graduate level course in Design Optimization. The problem is quite simple, yet the solution is not easily obtained. The beam consists of a specified number of equal length sections; each section has a rectangular cross section with the height constrained not to exceed 20 times the width. This equates to a crude buckling constraint. The maximum stress at the left end of each section is constrained as follows:

$$\sigma_i < +20 \text{ ksi} \qquad i = 1,5 \qquad (3.1)$$

The beam is cantilevered and tip loaded with a force of 10 kips downward and the total tip deflection is constrained not to exceed two inches. Material properties of the beam conform to steel where Young's modulus, $E = 30 \times 10^6$ psi.

The initial X vector of design variables consisting of height and width dimensions of each section is tabulated in Appendix A. There are five equal length segments in the overall length of 200 inches, resulting in an initial volume of 8000 in^3. The objective function is the minimum volume subject to the constraints of stress, displacement, and height to width ratio. A three dimensional drawing of the initial design is shown in Figure 3.1 and the optimum solution is shown in Figure 3.2. The tip deflection of the optimum beam is actually two inches downward but no attempt is made to show this in the figure.

2. <u>10-Bar Planar Truss</u>

Numerous test cases for planar trusses (2-dimensional) and space towers (3-dimensional) can be found in the literature. In particular, the 10-bar planar truss has been used in [Ref. 45] to demonstrate how the stress-ratio method, which seeks a fully-stressed design, yields poor results when members with significantly different allowable stresses are specified [Ref. 46].

The configuration of the 10-bar planar cantilever truss is shown in Figure 3.3 and is subject to a single load condition of 100 kips downward at nodes two and four. The initial cross-sectional areas of the truss elements and bounds on the areas are listed in Appendix B.

There are 20 constraints consisting of maximum and minimum stresses in each of the 10 members as follows:

$$-25 \text{ ksi} \leq \sigma_i \leq +25 \text{ ksi} \qquad i = 1\text{-}8,10 \qquad (3.2)$$

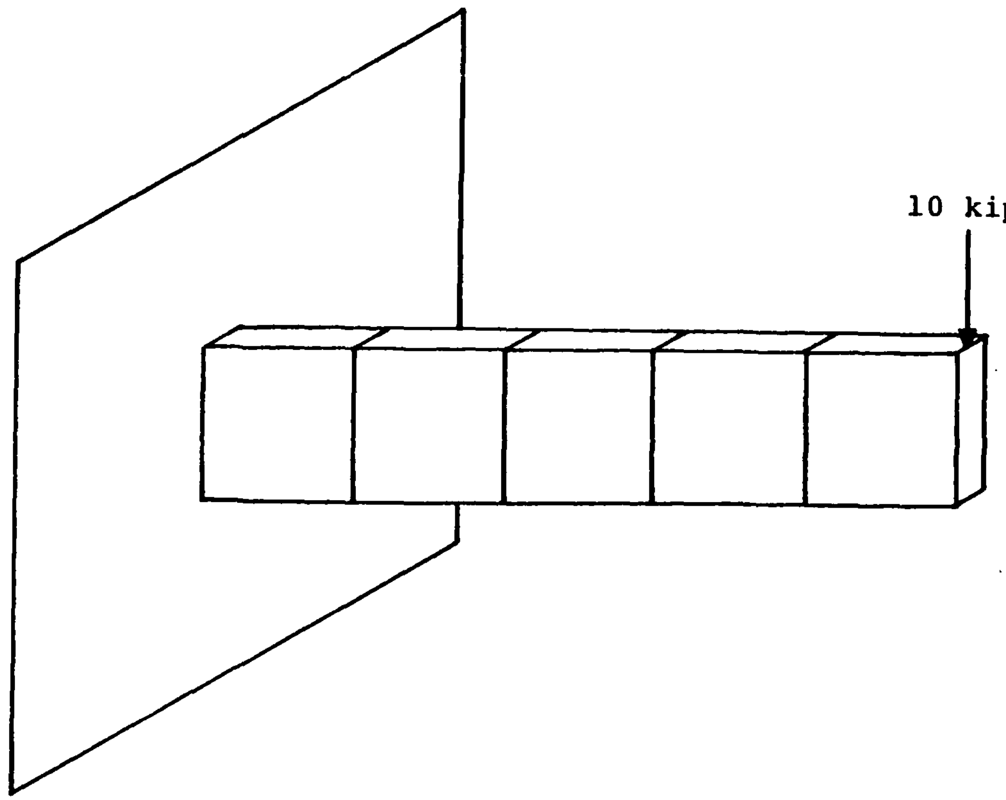


Figure 3.1. Initial Design of 10-Variable Cantilever Beam

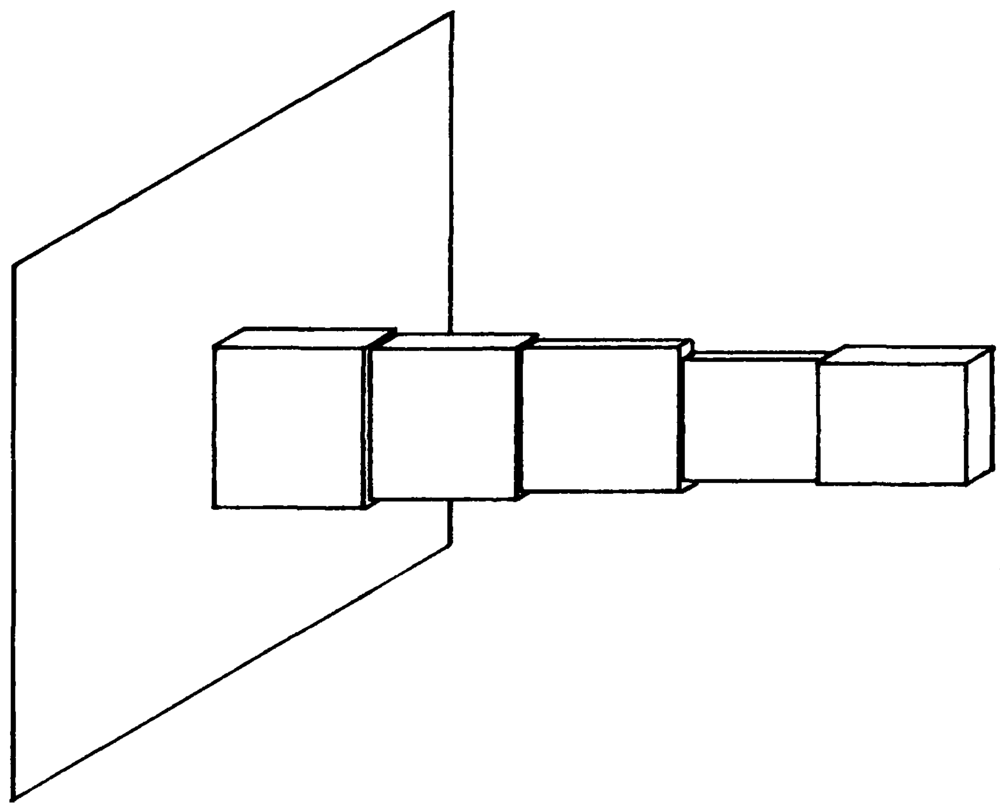


Figure 3.2. Optimum Design of 10-Variable Cantilever Beam

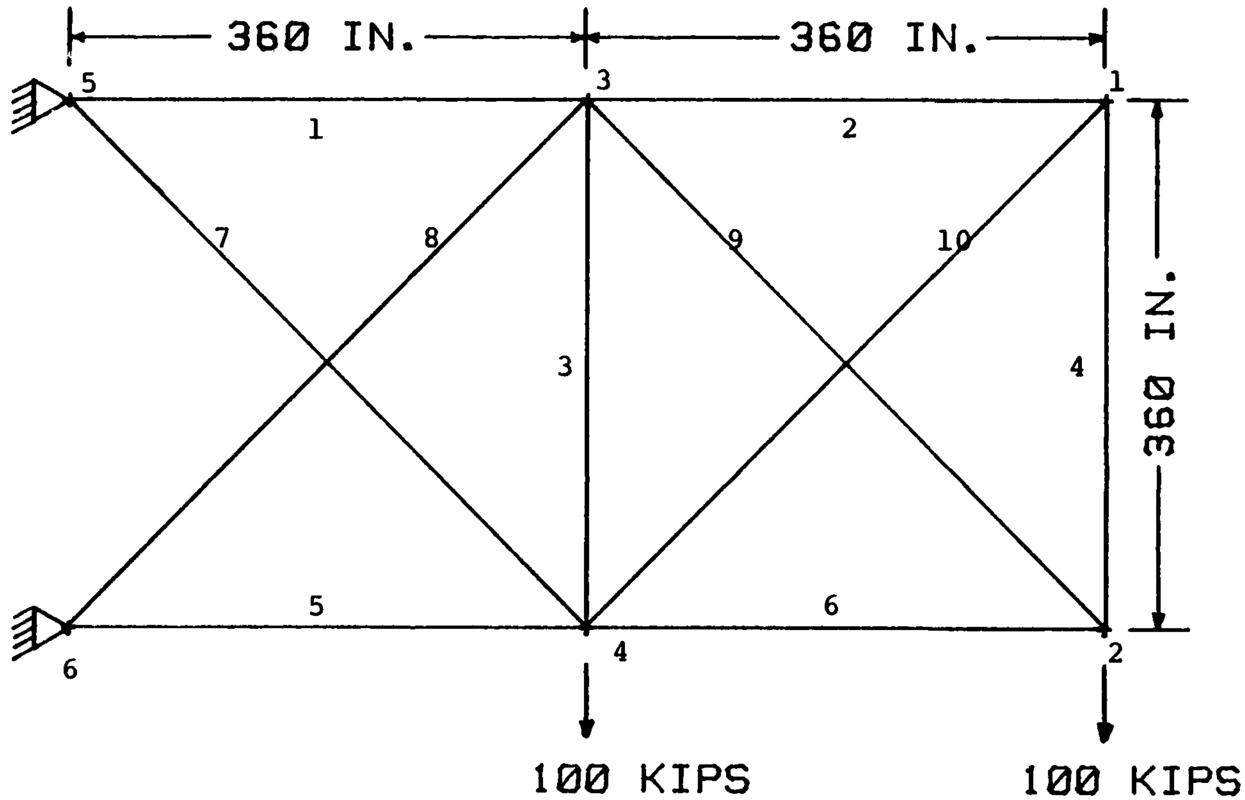


Figure 3.3. Configuration of the 10-Bar Planar Truss

$$-50 \text{ ksi} \leq \sigma_i \leq +50 \text{ ksi} \qquad i = 9 \qquad (3.3)$$

where i is the member element number. It should be noted that element nine in Figure 3.3 has twice the allowable stress of the other members. The objective function is minimum weight of the structure. Material properties include Young's Modulus, $E = 10 \times 10^6$ psi and $\gamma = .1$ lb/in^3 corresponding to the properties of aluminum.

3. Conceptual Design FFG-7 PERRY Class Frigate

The details of this test case may be obtained in [Ref. 47] where Jenkins optimized the conceptual design of a FFG-7 Perry Class Frigate. More specifically, he coupled the Reed synthesis model for surface combatant ships [Ref. 48], with the non-linear optimize CONMIN, a FORTRAN program for constrained function minimization, via the control program COPES. COPES/CONMIN was developed in 1973 by Vanderplaats [Ref. 5] and has been used in a variety of engineering applications. The objective function is the full load displacement of the vessel.

The design variables used in the preliminary testing of the ADS program are the same as those used by Jenkins: accordingly a comparison of results is appropriate. The independent design variables include:

1. LBP - Length between perpendiculars, ft.
2. L/B - Length to beam ratio
3. B/H - Beam to draft ratio

4. Cp - Prismatic coefficient

5. Cx - Midship section coefficient

The initial values of these variables as well as their upper and lower bounds are listed in Appendix C. There are 13 constraints on the design, these are explained in detail in [Ref. 47], and are not repeated here.

4. <u>47-Bar Planar Tower</u>

The 47-Bar planar tower shown in Figure 3.4 was introduced in the literature in [Ref. 49], wherein the tower was designed subject to multiple loading conditions. The same tower was designed for optimum geometry in [Ref. 50] subject to stress and Euler buckling. In [Ref. 50] sub-structuring was also used. The two sub-structures were over-lapped so that several members were in both sub-structures. [Ref. 51] presents configuration optimization with the addition of frequency constraints.

The 47-Bar planar tower used in this research is discussed in the remainder of this section. Initial cross-sectional areas of the truss elements, nodal coordinates and bounds on these parameters are tabulated in Appendix D as well as the details regarding displacement constraints and loading conditions. Steel was selected as the material for all members with Young's Modulus, $E = 30 \times 10^6$ psi and $\gamma = .3$ lb/in^3.

All elements are subject to the following constraints on stress:

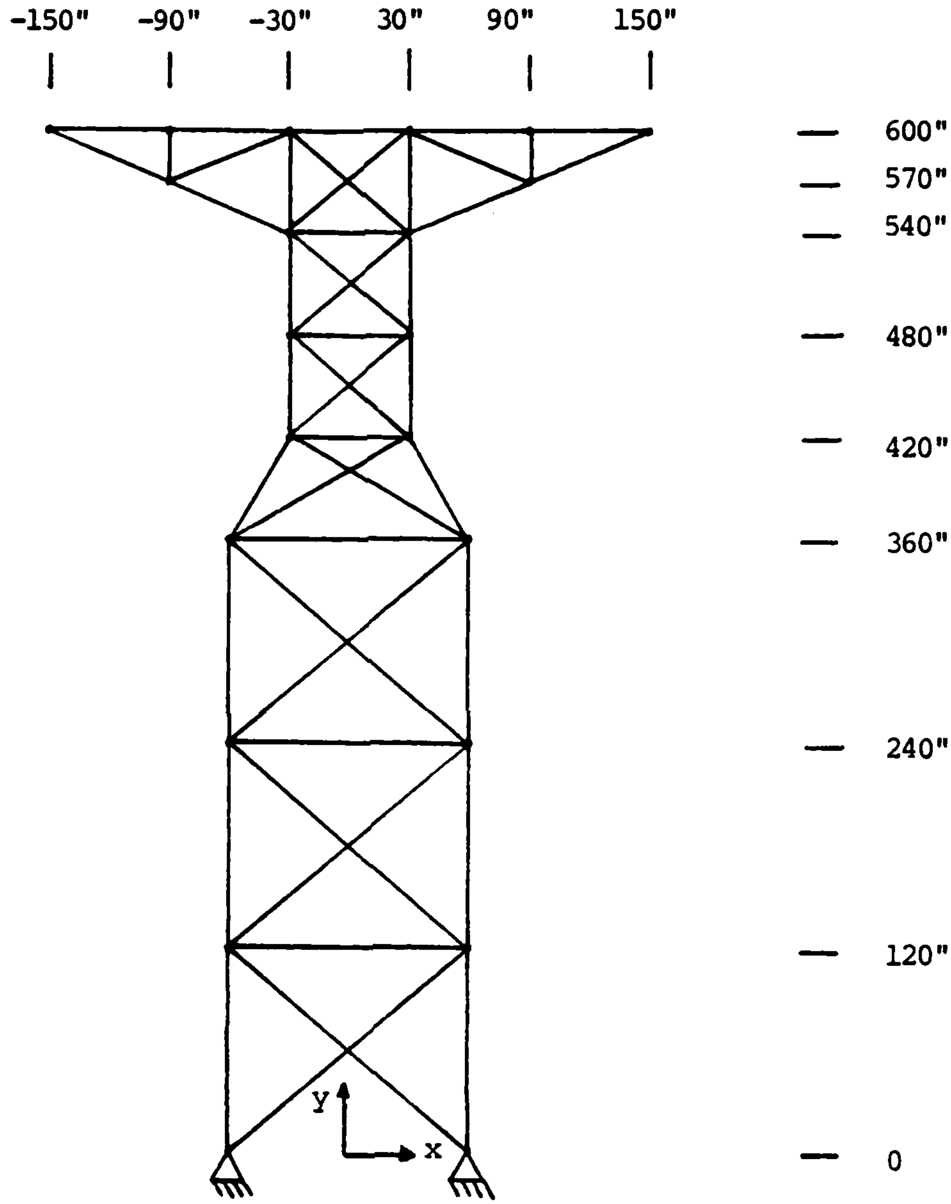


Figure 3.4. Initial Design of the 47-Bar Planar Tower

$$-15 \text{ ksi} \leq \sigma_i \leq +20 \text{ ksi} \qquad i = 1,47 \qquad (3.4)$$

where i is the element number. Tubular members are specified with a diameter to thickness (d/t) ratio = 10. Euler buckling is prohibited by constraining the buckling stress in the members according to the following equation:

$$\sigma_i \geq \sigma_{b_i} = -10.1\pi E A_i / 8L_i^2 \qquad i = 1,47 \qquad (3.5)$$

Finally, the first fundamental frequency of the structure is required to exceed 5. cps. Two non-structural weights of 500 lbs each are attached at nodes 17 and 22 to facilitate the eigenvalue problem solution.

Member areas and coordinates are linked to maintain symmetry about the vertical Y axis. Nodes 15, 16, 17 and 22 are fixed in space and nodes 1 and 2 are constrained to lie on the X axis. The resulting problem thus reduces to 27 member sizing variables and 17 configuration variables for a total of 44 independent design variables and 436 constraints on stress, Euler buckling, displacement and frequency.

The optimum design is shown in Figure 3.5. It should be noted there was no attempt to show member sizing variables in the figures.

5. <u>234-Bar Space Tower</u>

The configuration of the 234-Bar space tower is shown in Figure 3.6. Initial cross-sectional areas of the truss

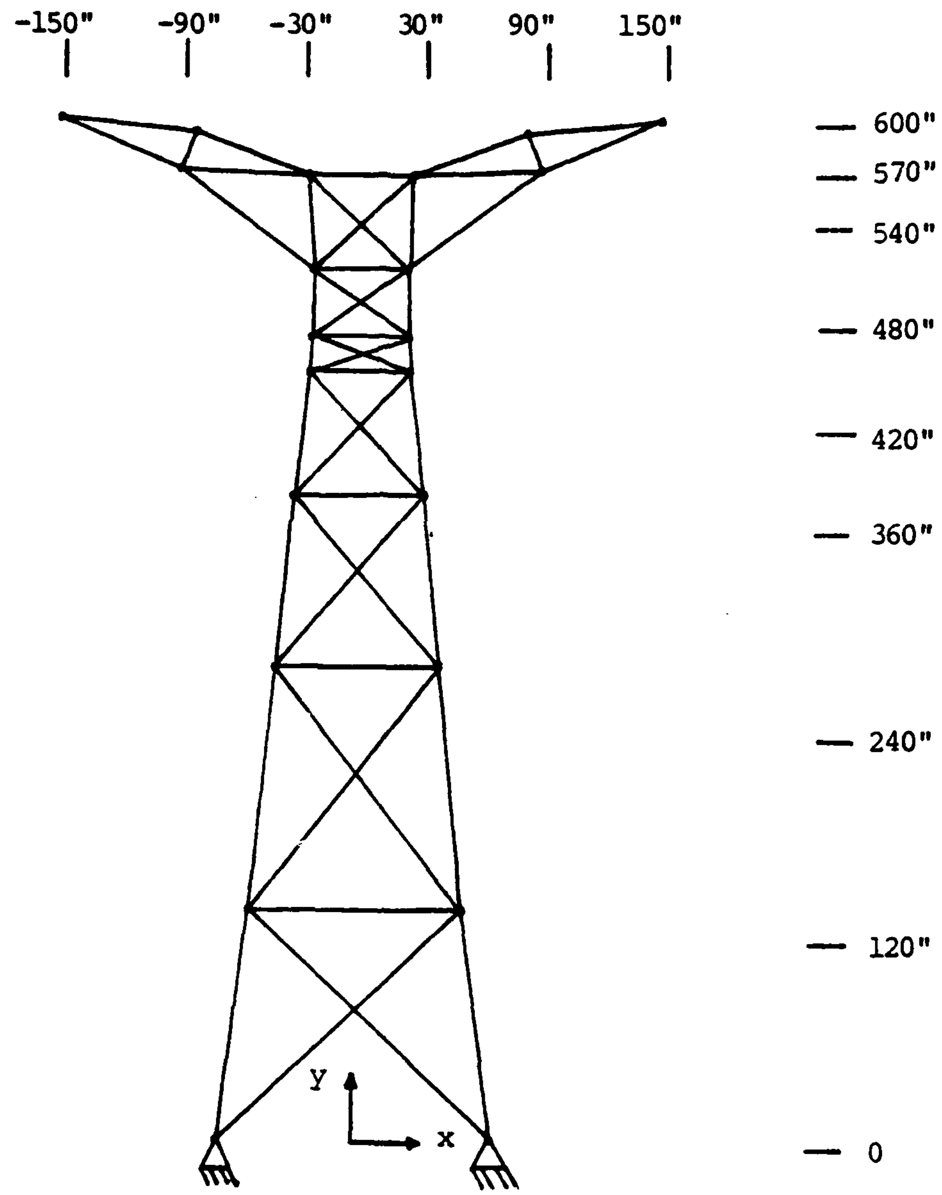


Figure 3.5. Optimum Design at the 47-Bar Planar Tower

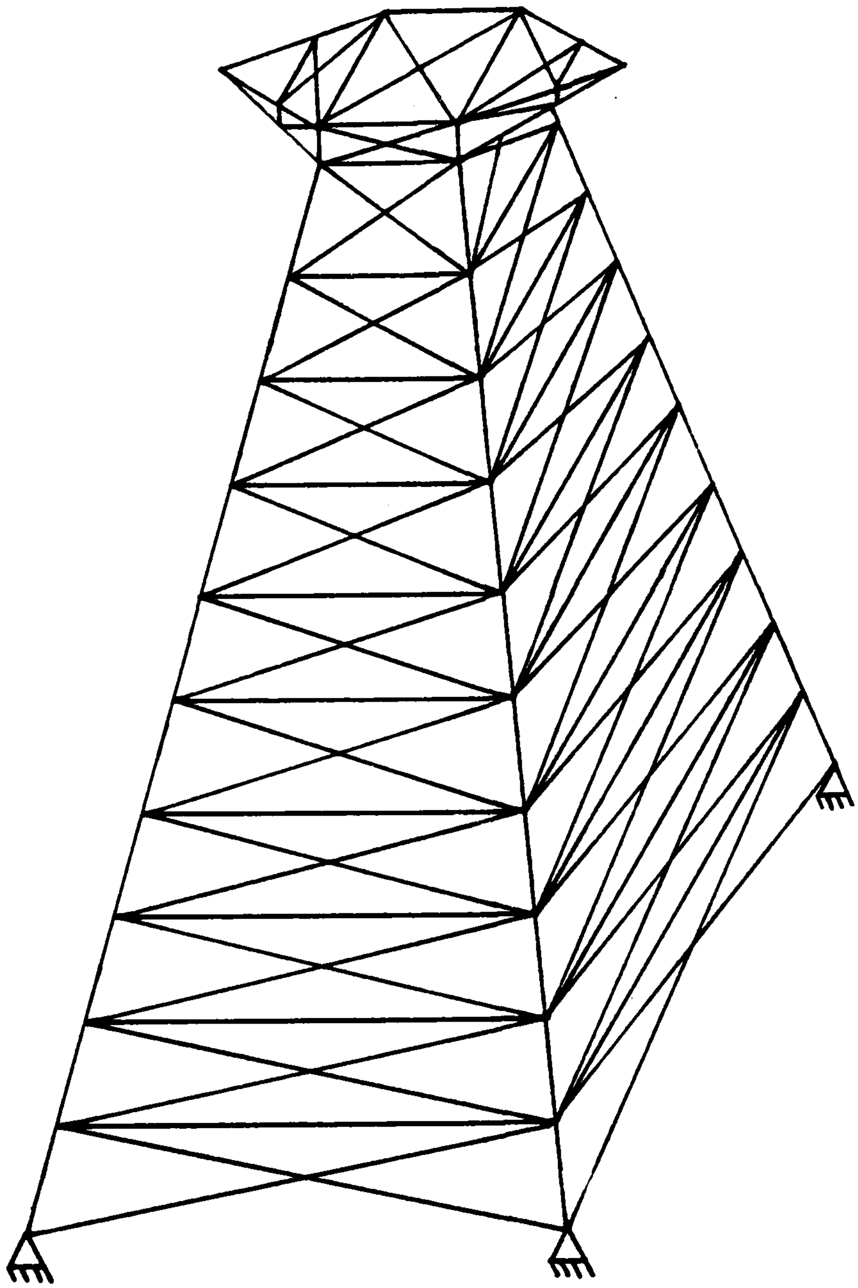


Figure 3.6. Configuration of the 234-Bar Space Tower

elements, nodal coordinates, bounds on these parameters, as well as the details regarding displacement constraints and loading conditions are tabulated in Appendix E. Aluminum was selected as the material for all members with Young's Modulus, $E = 10 \times 10^6$ psi and $\gamma = .1$ lb/in^3.

All elements are subject to the following constraints on stress:

$$-15 \text{ ksi} \leq \sigma_i \leq +20 \text{ ksi} \qquad i = 1{,}234 \tag{3.6}$$

where i is the element number. Tubular members are specified with a diameter to thickness (d/t) ratio = 10. Euler buckling is prohibited by constraining the buckling stress in the members according to the following equation:

$$\sigma_i > \sigma_{b_i} = -10.1\pi E A_i / 8 L_i^2 \qquad i = 1{,}234 \tag{3.7}$$

Member areas and coordinates are linked to maintain symmetry about the vertical Y axis. Nodes 1, 2, 3 and 4 are constrained to lie on the XZ plane. The resulting problem thus reduces to 56 member sizing variables and 3550 constraints on stress, Euler buckling, and displacement.

C. COUPLING ANALYSIS AND OPTIMIZATION COMPUTER CODES

The test case data files were prepared in accordance with the user's manual for SADT [Ref. 36] and the user's manual for COPESA, similar to [Ref. 6]. The problems were then coupled to the ADS library of optimization algorithms via a brief driver program in the case of trusses and towers and via COPESA on the cantilever beam and Ship design cases. All test case results were printed and filed for future reference. The default values for all program control parameters such as convergence tolerances were used insofar as possible. Gradients were calculated analytically for the truss and tower cases and by finite differences in the cases of the beam and ship.

The results obtained were carefully tabulated and optimum solutions determined based on the best objective function and the fewest equivalent function evaluations. This parameter was computed as follows:

$$NFE = IFCALL + NDV*IGCALL \tag{3.8}$$

where IFCALL is the number of objective and constraint function evaluations, IGCALL is the number of times gradients are evaluated by the user and NDV is the number of design variables. This provides an equivalent number of function evaluations that would be required if all gradients were calculated by finite differences. If gradients are calculated by finite differences, IGCALL will be zero because IFCALL includes the function evaluations needed to calculate gradients.

IV. RESULTS AND CONCLUSIONS

A. INTRODUCTION

There are presently 85 possible, meaningful combinations of strategy, optimizer and one-dimensional search methods available in the ADS library. Testing all methods on all problems is not practical considering some test cases consume over 40 minutes of CPU time per run. Accordingly, the scope of research was limited to testing all strategies and all optimizers with three one-dimensional searches on all five problems, for a total of 260 test case computer runs. The two one-dimensional search methods not tested were bounds only and polynomial without bounds. The results are tabulated in Appendix F. In Tables V through IX the best optimum designs to each of the five problems are presented. Optimum design A represents the best objective function achieved, whereas optimum design B represents the solution within 5% of the objective function for optimum design A but which had the fewest equivalent function evaluations. Both solutions were required to have no violated constraints ($g(X) < 0.01$).

B. RELATIVE RANKING OF OPTIMIZATION METHODS

1. Execution Time

A timing routine available in the Non-IMSL library at the computer center was utilized to record execution time in CPU seconds for each test run. Times were then averaged for all runs using the same one-dimensional search on a given problem. In other words, CPU time per function evaluation was averaged for all runs recorded on any given table in Appendix F. These run times, when multiplied by the equivalent number of function evaluations, is a good approximation of CPU seconds to optimize a problem with any given combination of strategy, optimizer and one-dimensional search. For example, average CPU time per function evaluation for the cantilever beam range from .002581 seconds to .0037283, whereas the range on the 234-Bar space tower is .32508 to .36011 seconds. It is readily apparent that on problems of significant size, like the 234-Bar tower (Table XXXV, Appendix F) run times of 34 CPU minutes may be realized. The significant point is that the efficiency of an algorithm to reduce NFE to a minimum is of vital concern on problems of practical interest.

2. <u>Number of Function/Gradient Calculations</u>

A perusal of all results in Appendix F reveals that direct methods are far more efficient than indirect methods as far as NFE is concerned when solving constrained minimization problems. Furthermore it is apparent that the ALM method is effective in reducing NFE for SUMT methods as theory would suggest.

Contrary to expectations, there is no apparent trend that would indicate which unconstrained minimization method is "best" to use when employing a SUMT method for the solution of a constrained problem. Perhaps more extensive testing would result in establishing these desirable guidelines.

A review of Table VII points out an interesting fact concerning NFE. In this table the optimum solutions for the FFG-7 test case are recorded. Note that the Method of Feasible Directions results in a quite acceptable objective function in 55 function evaluations while a SUMT method (exterior penalty) required 555 function evaluations to achieve a slightly better result! This situation is shown graphically in Figure 4.1. The point here is for the user to be aware of the possibility that an optimizer may be using an inordinate amount of computer resources to achieve an insignificant gain in the objective.

3. Values of the Objective Function

A comparison of objective functions points out that in general all presently available algorithms are working well in ADS with the exception of SUMT methods on the 234-Bar space tower. In this case the optimizers were unable to overcome the constraint violations and make progress toward a solution; the trouble is attributed to needed refinement in choosing the penalty parameters.

The efficiency, reliability, and accuracy of the various algorithms however, is clearly demonstrated on the other four test problems as recorded in Tables XXIII through XXXIV in Appendix F. In these four test cases, extremely good objective functions were obtained and generally resulted in the production of feasible designs (no violated constraints).

Tables V through IX record the best objective function achieved for each problem. Again, the direct methods

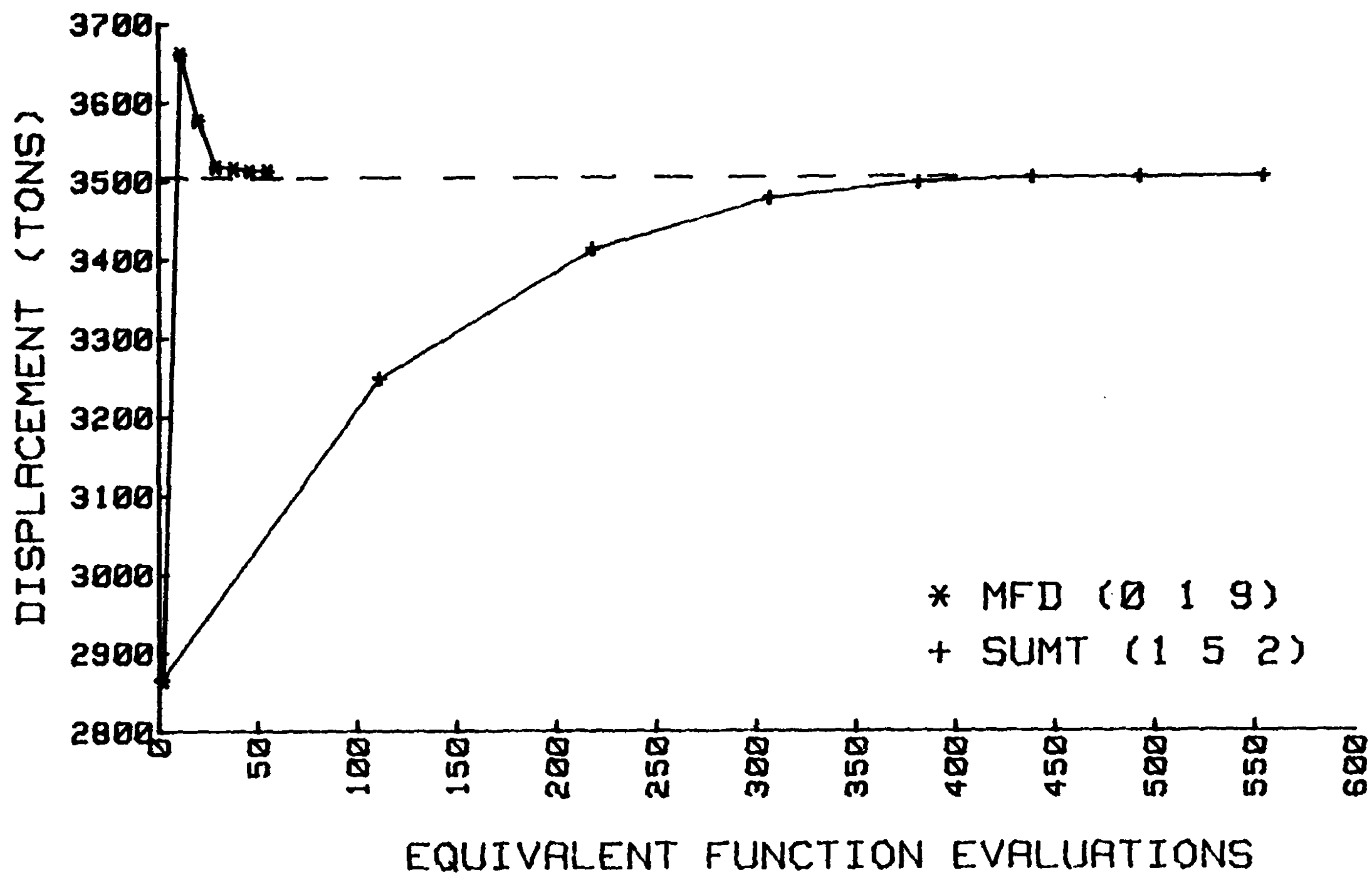


Figure 4.1. Iteration History for the FFG-7 Test Case

TABLE V

Optimum Design of 10-Variable Cantilever Beam

DESIGN VARIABLES	INITIAL VALUES	OPTIMUM DESIGN A	OPTIMUM DESIGN B
1	.20000E+01	.85538E+00	.85534E+00
2	.20000E+01	.84340E+00	.84342E+00
3	.20000E+01	.96538E+00	.96547E+00
4	.20000E+01	.10625E+01	.10626E+01
5	.20000E+01	.11446E+01	.11447E+01
6	.20000E+02	.17108E+02	.17107E+02
7	.20000E+02	.16868E+02	.16869E+02
8	.20000E+02	.19308E+02	.19310E+02
9	.20000E+02	.21251E+02	.21253E+02
10	.20000E+02	.22892E+02	.22894E+02
OBJ:	.80000E+04	.38513E+04	.38517E+04
	IFCALL:	365	191
	IGCALL:	9	0
	NFE:	365	191
	ISTRAT:	0	0
	IOPT:	3	3
	IONED:	8	9

TABLE VI

Optimum Design of 10-Bar Planar Truss

DESIGN VARIABLES	INITIAL VALUES	OPTIMUM DESIGN A	OPTIMUM DESIGN B
1	.10000E+02	.79149E+01	.78869E+01
2	.10000E+02	.10000E+00	.10000E+00
3	.10000E+02	.80888E+01	.81114E+01
4	.10000E+02	.39278E+01	.39154E+01
5	.10000E+02	.10000E+00	.10000E+00
6	.10000E+02	.10002E+00	.10001E+00
7	.10000E+02	.57843E+01	.58135E+01
8	.10000E+02	.54622E+01	.55145E+01
9	.10000E+02	.36814E+01	.36493E+01
10	.10000E+02	.14060E+00	.14060E+00
OBJ:	.41965E+04	.14955E+04	.14974E+04
	IFCALL:	219	76
	IGCALL:	10	14
	NFE:	319	216
	ISTRAT:	0	0
	IOPT:	3	3
	IONED:	7	9

TABLE VII

Optimum Conceptual Design of FFG-7 Perry Class Frigate

DESIGN VARIABLES	INITIAL VALUES	OPTIMUM DESIGN A	OPTIMUM DESIGN B
1	.30000E+03	.39429E+03	.39441E+03
2	.90700E+01	.73383E+01	.84923E+01
3	.31400E+01	.40000E+01	.34438E+01
4	.59300E+00	.51969E+00	.50000E+00
5	.75100E+00	.90000E+00	.77578E+00
OBJ:	.28650E+04	.35039E+04	.35120E+04
	IFCALL:	555	55
	IGCALL:	0	0
	NFE:	555	55
	ISTRAT:	1	0
	IOPT:	5	1
	IONED:	2	9

TABLE VIII

Optimum Design of 47-Bar Planar Tower

DESIGN VARIABLES	INITIAL VALUES	OPTIMUM DESIGN A	OPTIMUM DESIGN B
1	.50000E+01	.46860E+01	none within
2	.50000E+01	.41759E+01	5% of optimum
3	.50000E+01	.18816E+01	design A and
4	.50000E+01	.37585E+01	no violated
5	.50000E+01	.18577E+01	constraints
6	.50000E+01	.37921E+01	
7	.50000E+01	.32165E+01	
8	.50000E+01	.31278E+01	
9	.50000E+01	.21022E+01	
10	.50000E+01	.28899E+01	
11	.50000E+01	.22567E+01	
12	.50000E+01	.40493E+01	
13	.50000E+01	.24314E+01	
14	.50000E+01	.23419E+01	
15	.50000E+01	.37355E+01	
16	.50000E+01	.36838E+01	
17	.50000E+01	.42922E+01	
18	.50000E+01	.79178E+00	
19	.50000E+01	.29163E+01	
20	.50000E+01	.38352E+01	
21	.50000E+01	.94515E+00	
22	.50000E+01	.26998E+01	
23	.50000E+01	.41409E+01	
24	.50000E+01	.12333E+01	
25	.50000E+01	.24597E+01	
26	.50000E+01	.43808E+01	
27	.50000E+01	.10245E+01	
28	.60000E+02	.68591E+02	
29	.60000E+02	.53476E+02	
30	.12000E+03	.13530E+03	

TABLE VIII (Cont'd)

DESIGN VARIABLES	INITIAL VALUES	OPTIMUM DESIGN A	OPTIMUM DESIGN B
31	.60000E+02	.40992E+02	
32	.24000E+03	.27822E+03	
33	.60000E+02	.32738E+02	
34	.36000E+03	.37854E+03	
35	.30000E+02	.24938E+02	
36	.42000E+03	.45161E+03	
37	.30000E+02	.24265E+02	
38	.48000E+03	.47202E+03	
39	.30000E+02	.23896E+02	
40	.54000E+03	.51176E+03	
41	.90000E+02	.82690E+02	
42	.60000E+03	.59193E+03	
43	.30000E+02	.26367E+02	
44	.60000E+03	.56721E+03	
OBJ:	.28650E+04	.30141E+04	
	IFCALL:	948	
	IGCALL:	51	
	NFE:	3192	
	ISTRAT:	2	
	IOPT:	4	
	IONED:	2	

TABLE IX

Optimum Design of 234-Bar Space Tower

DESIGN VARIABLES	INITIAL VALUES	OPTIMUM DESIGN A	OPTIMUM DESIGN B
1	.25000E+02	.69559E+02	.69357E+02
2	.25000E+02	.68288E+02	.68117E+02
3	.25000E+02	.66756E+02	.66428E+02
4	.25000E+02	.64988E+02	.64421E+02
5	.25000E+02	.62568E+02	.61759E+02
6	.25000E+02	.59297E+02	.58494E+02
7	.25000E+02	.54735E+02	.53562E+02
8	.25000E+02	.48046E+02	.47554E+02
9	.25000E+02	.37844E+02	.37760E+02
10	.25000E+02	.21850E+02	.22198E+02
11	.25000E+02	.18359E+02	.18896E+02
12	.25000E+02	.20855E+00	.64660E+00
13	.25000E+02	.98045E-01	.40680E+00
14	.25000E+02	.16474E+00	.16005E+00
15	.25000E+02	.17637E+00	.22615E+00
16	.25000E+02	.96913E+00	.29182E+01
17	.25000E+02	.40538E+01	.60715E+01
18	.25000E+02	.72152E+01	.89224E+01
19	.25000E+02	.10221E+02	.11365E+02
20	.25000E+02	.12806E+02	.13766E+02
21	.25000E+02	.20106E+02	.20525E+02
22	.25000E+02	.21018E+02	.21433E+02
23	.25000E+02	.22353E+02	.22841E+02
24	.25000E+02	.25257E+02	.25643E+02
25	.25000E+02	.17203W+01	.19941E+01
26	.25000E+02	.16693E+01	.31480E+01
27	.25000E+02	.16977E+01	.21728E+01
28	.25000E+02	.22347E+01	.29806E+01
29	.25000E+02	.25725E+01	.25564E+01
30	.25000E+02	.30805E+01	.34746E+01
31	.25000E+02	.40618E+01	.35497E+01
32	.25000E+02	.58799E+01	.58570E+01

TABLE IX (Cont'd)

DESIGN VARIABLES	INITIAL VALUES	OPTIMUM DESIGN A	OPTIMUM DESIGN B
33	.25000E+02	.88958E+01	.86716E+01
34	.25000E+02	.13261E+02	.12863E+02
35	.25000E+02	.10249E+02	.10585E+02
36	.25000E+02	.19124E+02	.19571E+02
37	.25000E+02	.37666E+02	.38444E+02
38	.25000E+02	.15174E+02	.16716E+02
39	.25000E+02	.32889E+02	.33776E+02
40	.25000E+02	.20731E+02	.21057E+02
41	.25000E+02	.20853E+02	.21187E+02
42	.25000E+02	.29991E+02	.30891E+02
43	.25000E+02	.17611E+02	.18299E+02
44	.25000E+02	.95341E+01	.10795E+02
45	.25000E+02	.87553E+01	.10064E+02
46	.25000E+02	.12634E+00	.72552E-01
47	.25000E+02	.69780E-01	.20655E+01
48	.25000E+02	.22711E+01	.41014E+01
49	.25000E+02	.44747E+01	.61302E+01
50	.25000E+02	.66768E+01	.81517E+01
51	.25000E+02	.88756E+01	.10174E+02
52	.25000E+02	.11075E+02	.12196E+02
53	.25000E+02	.14990E+02	.15796E+02
54	.25000E+02	.15473E+02	.16240E+02
55	.25000E+02	.17706E+02	.18293E+02
56	.25000E+02	.17712E+02	.18299E+02
OBJ:	.84524E+05	.43967E+05	.45472E+05
	IFCALL:	1345	241
	IGCALL:	77	63
	NFE:	5657	3769
	ISTRAT:	5	5
	IOPT:	5	4
	IONED:	2	4

yielded a better result than the indirect methods on the two smaller problems while the SUMT methods prevailed on the larger problems. This suggests that indirect methods deal with a multitude of active constraints more effectively than direct methods.

C. COMPARISON WITH CONMIN

All test cases were run on CONMIN to provide a base-line for the comparative studies. Results from CONMIN are recorded on the tables of test results in Appendix F. It is interesting, if not surprising, that with the exception of the 234-bar tower test case, ADS routines were able to achieve better solutions than CONMIN.

CONMIN basically utilizes a Feasible Directions algorithm for constrained problems. The fact that a different combination surpassed CONMIN on each test case supports the notion that the optimization algorithm employed should suit the problem at hand to gain maximum efficiency. In the past the thrust has been to merely alter the program parameters of the same algorithm to deal with fundamentally different problems. ADS now offers a convenient method for selecting an algorithm best suited to the problem at hand. This flexibility further enhances an engineer's ability to apply optimization concepts to the various disciplines in design.

D. ADDITIONAL CONCLUSIONS

Preliminary testing of the ADS library resulted in the modifications of several default values for the various optimizers that improved their efficiency dramatically. As ADS is fully implemented additional testing will be required to insure all algorithms are as efficient, reliable and accurate as possible.

A difficulty with a program of this broad capability is to provide the user with a concise set of guidelines identifying which method or class of methods should be selected for a given problem. The problem is exacerbated by the selection of default values, in other words, a default value which may work well on one problem may cause premature convergence on a different problem. Accordingly, judicious selection of default values in ADS requires considerable effort supported by extensive testing on a variety of problems as the algorithms become operational.

Results given in Appendix F and the optimum solutions tabulated in Tables V through IX are an indication of reliability, to be sure, however the results are preliminary and the algorithms are constantly being revised and improved. The equivalent number of function evaluations (NFE) provide a measure of relative efficiency of the optimizer to achieve an optimum solution, the goal being to minimize the use of computer resources while maximizing the reduction of the objective function. It should be noted and is evident in the results tabulated that the efficiency and reliability are problem dependent. Therefore a wise selection of the appropriate algorithm and tailoring the program parameters to suit the problem at hand is required. ADS achieves this flexibility and enhances the design engineer's ability to use optimization as a viable design tool.

APPENDIX A

10-VARIABLE CANTILEVER BEAM TEST CASE

Table X describes the initial X vector of independent design variables; the first five variables are segment widths and the remaining five are segment heights; side constraints or bounds on the variables are also included.

TABLE X

Initial X Vector for the 10-Variable Cantilever Beam

	SEGMENT DIMENSIONS (INCHES)		
DESIGN VARIABLE	LOWER BOUND	INITIAL VALUE	UPPER BOUND
1	.50000E+00	.20000E+01	.50000E+01
2	.50000E+00	.20000E+01	.50000E+01
3	.50000E+00	.20000E+01	.50000E+01
4	.50000E+00	.20000E+01	.50000E+01
5	.50000E+00	.20000E+01	.50000E+01
6	.10000E+02	.20000E+02	.10000E+03
7	.10000E+02	.20000E+02	.10000E+03
8	.10000E+02	.20000E+02	.10000E+03
9	.10000E+02	.20000E+02	.10000E+03
10	.10000E+02	.20000E+02	.10000E+03

APPENDIX B

10-BAR PLANAR TRUSS TEST CASE

Table XI describes the initial X vector of independent design variables; the ten variables consist of the truss element cross-sectional areas. Side constraints or bounds on the variables are also included in the table. Table XII lists the nodal coordinates in inches.

TABLE XI

Initial X Vector for the 10-Bar Planar Truss

DESIGN VARIABLE NUMBER	TRUSS ELEMENT NUMBER	CROSS SECTIONAL AREAS (SQ. IN.)		
		LOWER BOUNDS	INITIAL VALUES	UPPER BOUNDS
1	1	.10000E+00	.10000E+02	.10000E+04
2	2	.10000E+00	.10000E+02	.10000E+04
3	3	.10000E+00	.10000E+02	.10000E+04
4	4	.10000E+00	.10000E+02	.10000E+04
5	5	.10000E+00	.10000E+02	.10000E+04
6	6	.10000E+00	.10000E+02	.10000E+04
7	7	.10000E+00	.10000E+02	.10000E+04
8	8	.10000E+00	.10000E+02	.10000E+04
9	9	.10000E+00	.10000E+02	.10000E+04
10	10	.10000E+00	.10000E+02	.10000E+04

TABLE XII

Initial Nodal Coordinates of the 10-Bar Planar Truss

NODE NUMBER	COORDINATES (INCHES) X	Y	Z
1	.72000E+03	.36000E+03	0
2	.72000E+03	0	0
3	.36000E+03	.36000E+03	0
4	.36000E+03	0	0
5	0	.36000E+03	0
6	0	0	0

APPENDIX C

CONCEPTUAL DESIGN OF THE FFG-7 PERRY CLASS FRIGATE

Table XIII lists the initial X vector of independent design variables; side constraints or bounds on the variables are included in the table.

TABLE XIII

Initial X Vector for the FFG-7 Preliminary Design

DESIGN VARIABLE NUMBER	PARAMETER	LOWER BOUNDS	INITIAL VALUES	UPPER BOUNDS
1	LBP	.30000E+03	.30000E+03	.70000E+03
2	L/B	.70000E+00	.90700E+00	.12000E+00
3	B/H	.20000E+01	.31400E+01	.40000E+01
4	Cp	.50000E+00	.59000E+00	.90000E+00
5	Cx	.75000E+00	.75000E+00	.90000E+00

APPENDIX D

47-BAR PLANAR TOWER TEST CASE

Tables XIV and XV describe the initial X vector of independent design variables. The 27 variables in Table XIV are the initial element cross-sectional areas. Table XV lists the initial nodal coordinates; 17 of these are independent design variables. It should be noted that since symmetry about the Y-axis exists only nodes on the positive side are listed. Side constraints or bounds on the variables are included in the tables. Tables XVI and XVII describe the loading conditions and displacement constraints respectively.

TABLE XIV

Initial Member Areas for the 47-Bar Planar Tower

DESIGN VARIABLE NUMBER	TRUSS ELEMENT NUMBER	CROSS SECTIONAL AREAS (SQ. IN.) LOWER BOUNDS	INITIAL VALUES	UPPER BOUNDS
1	3	.10000E-05	.50000E+01	.10000E+04
2	4	.10000E-05	.50000E+01	.10000E+04
3	5	.10000E-05	.50000E+01	.10000E+04
4	7	.10000E-05	.50000E+01	.10000E+04
5	8	.10000E-05	.50000E+01	.10000E+04
6	10	.10000E-05	.50000E+01	.10000E+04
7	12	.10000E-05	.50000E+01	.10000E+04
8	14	.10000E-05	.50000E+01	.10000E+04
9	15	.10000E-05	.50000E+01	.10000E+04
10	18	.10000E-05	.50000E+01	.10000E+04
11	20	.10000E-05	.50000E+01	.10000E+04
12	22	.10000E-05	.50000E+01	.10000E+04
13	24	.10000E-05	.50000E+01	.10000E+04
14	26	.10000E-05	.50000E+01	.10000E+04
15	27	.10000E-05	.50000E+01	.10000E+04
16	28	.10000E-05	.50000E+01	.10000E+04
17	30	.10000E-05	.50000E+01	.10000E+04
18	31	.10000E-05	.50000E+01	.10000E+04
19	33	.10000E-05	.50000E+01	.10000E+04
20	35	.10000E-05	.50000E+01	.10000E+04
21	36	.10000E-05	.50000E+01	.10000E+04
22	38	.10000E-05	.50000E+01	.10000E+04
23	40	.10000E-05	.50000E+01	.10000E+04
24	41	.10000E-05	.50000E+01	.10000E+04
25	43	.10000E-05	.50000E+01	.10000E+04
26	45	.10000E-05	.50000E+01	.10000E+04
27	46	.10000E-05	.50000E+01	.10000E+04

TABLE XV

Initial Nodal Coordinates of the 47-Bar Planar Tower

DESIGN VARIABLE NUMBER	NODE NUMBER	LOWER BOUNDS	COORDINATES (INCHES) X	Y	Z	UPPER BOUNDS
28	2	.1000E+02	.6000E+02	0	0	.1000E+04
29	4	.1000E+02	.6000E+02	0	0	.1000E+04
30	4	.1000E+02	0	.1200E+03	0	.1000E+04
31	6	.1000E+02	.6000E+02	0	0	.1000E+04
32	6	.1000E+02	0	.2400E+03	0	.1000E+04
33	8	.1000E+02	.6000E+02	0	0	.1000E+04
34	8	.1000E+02	0	.3600E+03	0	.1000E+04
35	10	.1000E+02	.3000E+02	0	0	.1000E+04
36	10	.1000E+02	0	.4200E+03	0	.1000E+04
37	12	.1000E+02	.3000E+02	0	0	.1000E+04
38	12	.1000E+02	0	.4800E+03	0	.1000E+04
39	14	.1000E+02	.3000E+02	0	0	.1000E+04
40	14	.1000E+02	0	.5400E+03	0	.1000E+04
	16		.9000E+02	.5700E+03	0	
41	20	.1000E+02	.3000E+02	0	0	.1000E+04
42	20	.1000E+02	0	.6000E+03	0	.1000E+04
43	21	.1000E+02	.9000E+02	0	0	.1000E+04
44	21	.1000E+02	0	.6000E+03	0	.1000E+04
	22		.1500E+03	.6000E+03	0	

TABLE XVI

Loading Conditions on the 47-Bar Planar Tower

LOAD CONDITION	NODE NUMBER	LOADS APPLIED (LBS) FX	FY	FZ
1	17	.6000E+04	-.14000E+05	0
	22	0	0	0
2	17	0	0	0
	22	.6000E+04	-.14000E+05	0
3	17	.6000E+04	-.14000E+05	0
	22	.6000E+04	-.14000E+05	0

TABLE XVII

Displacement Constraints on the 47-Bar Planar Tower

LOAD CONDITION	NODE NUMBER	DISPLACEMENT CONSTRAINTS (INCHES) DIRECTION	LOWER BOUNDS	UPPER BOUNDS
1	17	X	-.5000E+01	.5000E+01
	17	Y	-.5000E+01	.5000E+01
2	17	X	-.5000E+01	.5000E+01
	17	Y	-.5000E+01	.5000E+01
3	17	X	-.5000E+01	.5000E+01
	17	Y	-.5000E+01	.5000E+01

APPENDIX E

234-BAR SPACE TOWER TEST CASE

Table XVIII describes the initial X vector of independent design variables which consist of the member cross sectional areas. Side constraints or bounds on the variables are included in Table XVIII. Table XIX lists the nodal coordinates. Tables XX and XXI describe the loading conditions and displacement constraints respectively.

TABLE XVIII

Initial Member Areas for the 234-Bar Space Tower

DESIGN VARIABLE NUMBER	TRUSS ELEMENT NUMBER	CROSS SECTIONAL AREAS (SQ. IN.) LOWER BOUNDS	INITIAL VALUES	UPPER BOUNDS
1	1-4	.10000E-05	.25000E+02	.10000E+03
2	5-8	.10000E-05	.25000E+02	.10000E+03
3	9-12	.10000E-05	.25000E+02	.10000E+03
4	13-16	.10000E-05	.25000E+02	.10000E+03
5	17-20	.10000E-05	.25000E+02	.10000E+03
6	21-24	.10000E-05	.25000E+02	.10000E+03
7	25-28	.10000E-05	.25000E+02	.10000E+03
8	29-32	.10000E-05	.25000E+02	.10000E+03
9	33-36	.10000E-05	.25000E+02	.10000E+03
10	37-40	.10000E-05	.25000E+02	.10000E+03
11	41-44	.10000E-05	.25000E+02	.10000E+03
12	45-48	.10000E-05	.25000E+02	.10000E+03
13	49-52	.10000E-05	.25000E+02	.10000E+03
14	53-56	.10000E-05	.25000E+02	.10000E+03
15	57-60	.10000E-05	.25000E+02	.10000E+03
16	61-64	.10000E-05	.25000E+02	.10000E+03
17	65-68	.10000E-05	.25000E+02	.10000E+03
18	69-72	.10000E-05	.25000E+02	.10000E+03
19	73-76	.10000E-05	.25000E+02	.10000E+03
20	77-80	.10000E-05	.25000E+02	.10000E+03
21	81,83	.10000E-05	.25000E+02	.10000E+03
22	82,84	.10000E-05	.25000E+02	.10000E+03
23	85,87	.10000E-05	.25000E+02	.10000E+03
24	86,88	.10000E-05	.25000E+02	.10000E+03
25	89-96	.10000E-05	.25000E+02	.10000E+03
26	97-104	.10000E-05	.25000E+02	.10000E+03
27	105-112	.10000E-05	.25000E+02	.10000E+03
28	113-120	.10000E-05	.25000E+02	.10000E+03
29	121-128	.10000E-05	.25000E+02	.10000E+03
30	129-136	.10000E-05	.25000E+02	.10000E+03

TABLE XVIII (Cont'd)

DESIGN VARIABLE NUMBER	TRUSS ELEMENT NUMBER	CROSS SECTIONSL AREAS (SQ. IN.) LOWER BOUNDS	INITIAL VALUES	UPPER BOUNDS
31	137-144	.10000E-05	.25000E+02	.10000E+03
32	145-152	.10000E-05	.25000E+02	.10000E+03
33	153-160	.10000E-05	.25000E+02	.10000E+03
34	161-168	.10000E-05	.25000E+02	.10000E+03
35	169-176	.10000E-05	.25000E+02	.10000E+03
36	177-180	.10000E-05	.25000E+02	.10000E+03
37	181-184	.10000E-05	.25000E+02	.10000E+03
38	185-188	.10000E-05	.25000E+02	.10000E+03
39	189-192	.10000E-05	.25000E+02	.10000E+03
40	193-196	.10000E-05	.25000E+02	.10000E+03
41	197-200	.10000E-05	.25000E+02	.10000E+03
42	201,202	.10000E-05	.25000E+02	.10000E+03
43	203,204	.10000E-05	.25000E+02	.10000E+03
44	205-208	.10000E-05	.25000E+02	.10000E+03
45	209-212	.10000E-05	.25000E+02	.10000E+03
46	213,214	.10000E-05	.25000E+02	.10000E+03
47	215,216	.10000E-05	.25000E+02	.10000E+03
48	217,218	.10000E-05	.25000E+02	.10000E+03
49	219,220	.10000E-05	.25000E+02	.10000E+03
50	221,222	.10000E-05	.25000E+02	.10000E+03
51	223,224	.10000E-05	.25000E+02	.10000E+03
52	225,226	.10000E-05	.25000E+02	.10000E+03
53	227,228	.10000E-05	.25000E+02	.10000E+03
54	229,230	.10000E-05	.25000E+02	.10000E+03
55	231,232	.10000E-05	.25000E+02	.10000E+03
56	233,234	.10000E-05	.25000E+02	.10000E+03

TABLE XIX

Nodal Coordinates of the 234-Bar Space Tower

NODE NUMBER	COORDINATES (INCHES) X	Y	Z
1	.12000E+03	0	.12000E+03
2	-.12000E+03	0	.12000E+03
3	-.12000E+03	0	-.12000E+03
4	.12000E+03	0	-.12000E+03
5	.11100E+03	.12000E+03	.11100E+03
6	-.11100E+03	.12000E+03	.11100E+03
7	-.11100E+03	.12000E+03	-.11100E+03
8	.11100E+03	.12000E+03	-.11100E+03
9	.10200E+03	.24000E+03	.10200E+03
10	-.10200E+03	.24000E+03	-.10200E+03
11	-.10200E+03	.24000E+03	-.10200E+03
12	.10200E+03	.24000E+03	.10200E+03
13	.93000E+02	.36000E+03	.93000E+02
14	-.93000E+02	.36000E+03	.93000E+02
15	-.93000E+02	.36000E+03	-.93000E+02
16	.93000E+02	.36000E+03	-.93000E+02
17	.84000E+02	.48000E+03	.84000E+02
18	-.84000E+02	.48000E+03	.84000E+02
19	-.84000E+02	.48000E+03	-.84000E+02
20	.84000E+02	.48000E+03	-.84000E+02
21	.75000E+02	.60000E+03	.75000E+02
22	-.75000E+02	.60000E+03	.75000E+02
23	-.75000E+02	.60000E+03	-.75000E+02
24	.75000E+02	.60000E+03	-.75000E+02
25	.66000E+02	.72000E+03	.66000E+02
26	-.66000E+02	.72000E+03	.66000E+02
27	-.66000E+02	.72000E+03	-.66000E+02
28	.66000E+02	.72000E+03	-.66000E+02
29	.57000E+02	.84000E+03	.57000E+02
30	-.57000E+02	.84000E+03	.57000E+02

TABLE XIX (Cont'd)

NODE NUMBER	COORDINATES (INCHES) X	Y	Z
31	-.57000E+02	.84000E+03	-.57000E+02
32	.57000E+02	.84000E+03	-.57000E+02
33	.48000E+02	.96000E+03	.48000E+02
34	-.48000E+02	.96000E+03	.48000E+02
35	-.48000E+02	.96000E+03	-.48000E+02
36	.48000E+02	.96000E+03	-.48000E+02
37	.39000E+02	.10800E+04	.39000E+02
38	-.39000E+02	.10800E+04	.39000E+02
39	-.39000E+02	.10800E+04	-.39000E+02
40	.39000E+02	.10800E+04	-.39000E+02
41	.30000E+02	.12000E+04	.30000E+02
42	-.30000E+02	.12000E+04	.30000E+02
43	-.30000E+02	.12000E+04	-.30000E+02
44	.30000E+02	.12000E+04	-.30000E+02
45	.30000E+02	.12480E+04	.30000E+02
46	-.30000E+02	.12480E+04	.30000E+02
47	-.30000E+02	.12480E+04	-.30000E+02
48	.30000E+02	.12480E+04	-.30000E+02
49	.90000E+02	.12480E+04	0
50	-.90000E+02	.12480E+04	0
51	.60000E+02	.12240E+04	.15000E+02
52	-.60000E+02	.12240E+04	.15000E+02
53	.60000E+02	.12480E+04	.15000E+02
54	-.60000E+02	.12480E+04	.15000E+02
55	-.60000E+02	.12240E+04	-.15000E+02
56	.60000E+02	.12240E+04	-.15000E+02
57	-.60000E+02	.12480E+04	-.15000E+02
58	.60000E+02	.12480E+04	-.15000E+02

TABLE XX

Loading Conditions on the 234-Bar Space Tower

LOAD CONDITION	NODE NUMBER	LOADS APPLIED (LBS) FX	FY	FZ
1	49	.6000E+04	-.20000E+05	0
	50	.6000E+04	-.20000E+05	0
2	49	.6000E+04	-.20000E+05	0
	50	-.6000E+04	-.20000E+05	0
3	49	.6000E+04	-.20000E+05	0
	50	.3000E+04	-.10000E+05	.50000E+04
4	49	.3000E+04	-.10000E+05	-.50000E+04
	50	.3000E+04	-.10000E+05	.50000E+04
5	49	-.3000E+04	.10000E+05	.50000E+04
	50	-.3000E+04	.10000E+05	-.50000E+04

TABLE XXI

Displacement Constraints on the 234-Bar Space Tower

LOAD CONDITION	NODE NUMBER	DISPLACEMENT CONSTRAINTS (INCHES) DIRECTION	LOWER BOUNDS	UPPER BOUNDS
1	49	X,Y	-.5000E+01	.5000E+01
	50	X,Y	-.5000E+01	.5000E+01
2	49	X,Y	-.5000E+01	.5000E+01
	50	X,Y	-.5000E+01	.5000E+01
3	49	X,Y	-.5000E+01	.5000E+01
	50	X,Y	-.5000E+01	.5000E+01
4	49	X,Y	-.5000E+01	.5000E+01
	50	X,Y	-.5000E+01	.5000E+01
5	49	X,Y	-.5000E+01	.5000E+01
	50	X,Y	-.5000E+01	.5000E+01

APPENDIX F

ADS-1 PRELIMINARY TEST RESULTS

The results of the preliminary testing of the algorithms available in version 1 of the ADS library in February 1983 are summarized in Tables XXIII through XXXVII. The nomenclature used in these tables is defined in Table XXII.

TABLE XXII

Definition of Terms in Test Results

TERM	DESCRIPTION
NDV	Number of independent design variables
NCON	Number of constraints on design
OBJ	Objective function value
NAC	Number of active constraints
NFE	Number of Equivalent Function Evaluations
SUMT	Sequential Unconstrained Minimization Technique
EXT	Exterior Penalty Method
LIN EXT INT	Linear Extended Interior Penalty Method
QUAD EXT INT	Quadratic Extended Interior Penalty Method
CUBIC EXT INT	Cubic Extended Interior Penalty Method
ALM	Augmented Lagrange Multipliers Method
MFD	Method of Feasible Directions
DFP	Davidon-Fletcher-Powell Algorithm
BFGS	Broydon-Fletcher-Goldfarb-Shanno Algorithm

TABLE XXIII

Test Results 10-Variable Cantilever Beam (IONED: 2,7)

ONE-DIMENSIONAL SEARCH: Golden Section

NDV = 10   NCON = 11 (Stress, Displacement and H/B Ratio)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .27097E-02 seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.47264E+04		.38513E+04		
DIRECT	NAC	1		10		
	NFE	594		532		
1	OBJ		.39742E+04		.39188E+04	.39067E+04
SUMT	NAC		1*		2*	4
(EXT)	NFE		568		691	738
2 SUMT	OBJ		.47597E+04		.39268E+04	.39591E+04
(LIN-	NAC		3		2	3
EXT-INT)	NFE		736		777	795
3 SUMT	OBJ		.47597E+04		.39288E+04	.39591E+04
(QUAD-	NAC		3		1	3
EXT-INT)	NFE		736		772	795
4 SUMT	OBJ		.47597E+04		.39241E+04	.39591E+04
(CUBIC	NAC		3		3	3
EXT-INT)	NFE		736		892	795
5 SUMT	OBJ		.38793E+04		.38968E+04	.38961E+04
(ALM)	NAC		4*		0*	0*
	NFE		922		692	799

CONMIN RESULTS:
OBJ: .40808E+04
NAC: 4
NFE: 313

* = VIOLATED CONSTRAINT(S)

TABLE XXIV

Test Results 10-Variable Cantilever Beam (IONED: 3,8)

ONE-DIMENSIONAL SEARCH: Golden Section + Cubic Polynomial

NDV = 10 NCON = 11 (Stress, Displacement and H/B Ratio)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .25810E-02 Seconds

OPTIMIZER: STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.49338E+04		.38513E+04		
DIRECT	NAC	1		10		
	NFE	403		365		
1	OBJ		.39856E+04		.40222E+04	.38984E+04
SUMT	NAC		1*		5	1*
(EXT)	NFE		440		611	501
2 SUMT	OBJ		.47397E+04		.39563E+04	.39208E+04
(LIN-	NAC		3		3	1
EXT-	NFE		549		594	599
INT)						
3 SUMT	OBJ		.47394E+04		.39567E+04	.39211E+04
(QUAD-	NAC		3		3	1
EXT-	NFE		552		599	599
INT)						
4 SUMT	OBJ		.47395E+04		.39566E+04	.39211E+04
(CUBIC	NAC		3		3	1
EXT-	NFE		552		599	599
INT)						
5	OBJ		.39740E+04		.38657E+04	.38654E+04
SUMT	NAC		2*		2*	4*
(ALM)	NFE		770		587	766

CONMIN RESULTS:
OBJ: .40808E+04
NAC: 4
NFE: 313

* = VIOLATED CONSTRAINT(S)

TABLE XXV

Test Results 10-Variable Cantilever Beam (IONED: 4,9)

ONE-DIMENSIONAL SEARCH: Bounds + Polynomial

NDV = 10   NCON = 11 (Stress, Displacement and H/B Ratio)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .37283E-02 Seconds

OPTIMIZER STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.49772E+04		.38517E+04		
DIRECT	NAC	1		10		
	NFE	287		191		
1	OBJ		.39589E+04		.39489E+04	.39849E+04
SUMT	NAC		2		1*	0*
(EXT)	NFE		397		333	293
2						
SUMT	OBJ		.50299E+04		.39495E+04	.39655E+04
(LIN-	NAC		1		2	3
EXT-	NFE		342		490	519
INT)						
3						
SUMT	OBJ		.49392E+04		.39313E+04	.39689E+04
(QUAD-	NAC		1		2	2
EXT-	NFE		400		388	434
INT)						
4						
SUMT	OBJ		.49982E+04		.39260E+04	.39562E+04
(CUBIC	NAC		1		2	1
EXT-	NFE		369		436	411
INT)						
5	OJB		.39678E+04		.38536E+04	.38580E+04
SUMT	NAC		4		2*	3*
(ALM)	NFE		531		396	461

CONMIN RESULTS:
OBJ: .40808E+04
NAC: 4
NFE: 313

* = VIOLATED CONSTRAINT(S)

TABLE XXVI

Test Results 10-Bar Planar Truss (IONED: 2,7)

ONE-DIMENSIONAL SEARCH: Golden Section

NDV = 10 NCON = 20 (Stress)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .41632E-02 Seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0 DIRECT	OBJ	.15436E+04		.14955E+04		
	NAC	1		8		
	NFE	677		319		
1 SUMT (EXT)	OBJ		.16708E+04		.15558E+04	.15372E+04
	NAC		6		5	8
	NFE		860		824	967
2 SUMT (LIN-EXT-INT)	OBJ		.16205E+04		.15104E+04	.15750E+04
	NAC		3		6	5
	NFE		1137		1329	1351
3 SUMT (QUAD-EXT-INT)	OBJ		.16205E+04		.15967E+04	.15750E+04
	NAC		3		4	5
	NFE		1137		1356	1351
4 SUMT (CUBIC EXT-INT)	OBJ		.16205E+04		.15967E+04	.15750E+04
	NAC		3		4	5
	NFE		1137		1356	1351
5 SUMT (ALM)	OBJ		.15733E+04		.15285E+04	.14987E+04
	NAC		7		8	8
	NFE		1138		1091	1389

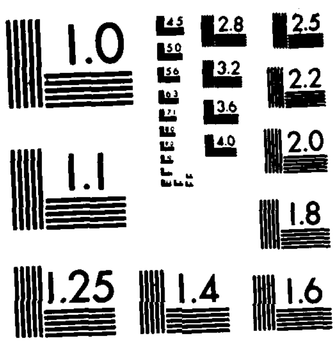
CONMIN RESULTS:
OBJ: .15009E+04
NAC: 10
NFE: 414

* = VIOLATED CONSTRAINT(S)

AD-A128 609 A COMPARATIVE STUDY OF OPTIMIZATION ALGORITHMS FOR ENGINEERING SYNTHESIS(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA C M SPRAGUE MAR 83 2/2

UNCLASSIFIED F/G 9/2 NL

END
FILMED



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

TABLE XXVII

Test Results 10-Bar Planar Truss (IONED: 3,8)

ONE-DIMENSIONAL SEARCH: Golden Section + Cubic Polynomial

NDV = 10   NCON = 20 (Stress)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .39329E-02 Seconds

OPTIMIZER / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.15445E+04		.14960E+04		
DIRECT	NAC	1		8		
	NFE	666		244		
1	OBJ		.16717E+04		.15743E+04	.15390E+04
SUMT	NAC		6		6	7
(EXT)	NFE		622		721	681
2 SUMT	OBJ		.16160E+04		.15103E+04	.16357E+04
(LIN-	NAC		3		5	2
EXT- INT)	NFE		885		1059	856
3 SUMT	OBJ		.16202E+04		.15760E+04	.15684E+04
(QUAD-	NAC		3		5	5
EXT- INT)	NFE		826		1002	902
4 SUMT	OBJ		.16185E+04		.15747E+04	.16352E+04
(CUBIC	NAC		3		5	4
EXT- INT)	NFE		936		1137	1016
5	OBJ		.16076E+04		.14996E+04	.14996E+04
SUMT	NAC		7		8	8
(ALM)	NFE		786		868	812

CONMIN RESULTS:
OBJ: .15009E+04
NAC: 10
NFE: 414

* = VIOLATED CONSTRAINT(S)

TABLE XXVIII

Test Results 10-Bar Planar Truss (IONED: 4,9)

ONE-DIMENSIONAL SEARCH: Bounds + Polynomial

NDV = 10 NCON = 20 (Stress)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .43674E-02 Seconds

OPTIMIZER STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.15313E+04		.14974E+04		
DIRECT	NAC	1		8		
	NFE	471		216		
1	OBJ		.16488E+04		.16755E+04	.15752E+04
SUMT	NAC		6		7	8
(EXT)	NFE		384		310	509
2						
SUMT	OBJ		.16034E+04		.15102E+04	.15095E+04
(LIN-	NAC		3		3	7
EXT-	NFE		676		691	807
INT)						
3						
SUMT	OBJ		.16809E+04		.15113E+04	.15100E+04
(QUAD-	NAC		4		4	5
EXT-	NFE		525		721	831
INT)						
4						
SUMT	OBJ		.16822E+04		.15104E+04	.15008E+04
(CUBIC	NAC		4		4	8
EXT-	NFE		525		812	729
INT)						
5	OBJ		.15265E+04		.15414E+04	.15121E+04
SUMT	NAC		4*		6*	8
(ALM)	NFE		423		446	780

CONMIN RESULTS:
OBJ: .15009E+04
NAC: 10
NFE: 414

* = VIOLATED CONSTRAINT(S)

TABLE XXIX

Test Results Conceptual Design FFG-7 (IONED: 2,7)

ONE-DIMENSIONAL SEARCH: Golden Section

NDV = 5 NCON = 13

AVERAGE CPU TIME PER FUNCTION EVALUATION: .23774E-01 Seconds

OPTIMIZER: STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0 DIRECT	OBJ	.35122E+04		.35077E+04		
	NAC	1		2		
	NFE	155		170		
1 SUMT (EXT)	OBJ		.35114E+04		.35044E+04	.35039E+04
	NAC		1		2	2
	NFE		511		514	555
2 SUMT (LIN-EXT-INT)	OBJ		.35224E+04		.35060E+04	.35067E+04
	NAC		1		1	1
	NFE		502		544	604
3 SUMT (QUAD-EXT-INT)	OBJ		.35265E+04		.35087E+04	.35052E+04
	NAC		1		1	1
	NFE		504		537	597
4 SUMT (CUBIC EXT-INT)	OBJ		.35260E+04		.35122E+04	.35088E+04
	NAC		1		1	1
	NFE		503		514	535
5 SUMT (ALM)	OBJ		.35107E+04		.35063E+04	.35082E+04
	NAC		1		0*	2
	NFE		285		284	248

CONMIN RESULTS:

OBJ: .35128E+04

NAC: 3

NFE: 60

* = VIOLATED CONSTRAINT(S)

TABLE XXX

Test Results Conceptual Design FFG-7 (IONED: 3,8)

ONE-DIMENSIONAL SEARCH: Golden Section + Cubic Polynomial

NDV = 5   NCON = 13

AVERAGE CPU TIME PER FUNCTION EVALUATION: .25550E-01 Seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.35112E+04		.35079E+04		
DIRECT	NAC	0		2		
	NFE	117		107		
1	OBJ		.35110E+04		.35039E+04	.35043E+04
SUMT	NAC		1		2	2
(EXT)	NFE		374		388	393
2						
SUMT	OBJ		.35235E+04		.35088E+04	.35089E+04
(LIN-	NAC		1		1	1
EXT-	NFE		359		366	368
INT)						
3						
SUMT	OBJ		.35206E+04		.35088E+04	.35091E+04
(QUAD-	NAC		1		1	1
EXT-	NFE		352		356	351
INT)						
4						
SUMT	OBJ		.35261E+04		.35082E+04	.35087E+04
(CUBIC	NAC		1		1	1
EXT-	NFE		354		399	358
INT)						
5	OBJ		.35130E+04		.35065E+04	.35073E+04
SUMT	NAC		0*		1*	2
(ALM)	NFE		192		204	210

CONMIN RESULTS:

OBJ: .35128E+04

NAC: 3

NFE: 60

* = VIOLATED CONSTRAINT(S)

TABLE XXXI

Test Results Conceptual Design FFG-7 (IONED: 4,9)

ONE-DIMENSIONAL SEARCH: Bounds + Polynomial

NDV = 5 NCON = 13

AVERAGE CPUT TIME PER FUNCTION EVALUATION: .28926E-01 Seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.35120E+04		.35078E+04		
DIRECT	NAC	1		2		
	NFE	55		78		
1	OBJ		.35109E+04		.35085E+04	.35083E+04
SUMT	NAC		1		2	2
(EXT)	NFE		253		262	260
2 SUMT	OBJ		.35216E+04		.35174E+04	.35150E+04
(LIN-	NAC		0		1	1
EXT-INT)	NFE		238		242	242
3 SUMT	OBJ		.35305E+04		.35084E+04	.35091E+04
(QUAD-	NAC		1		1	1
EXT-INT)	NFE		227		227	226
4 SUMT	OBJ		.35135E+04		.35090E+04	.35090E+04
(CUBIC	NAC		1		1	1
EXT-INT)	NFE		228		228	222
5	OBJ		.35086E+04		.35085E+04	.35095E+04
SUMT	NAC		0*		1	1
(ALM)	NFE		130		169	174

CONMIN RESULTS:
OBJ: .35128E+04
NAC: 3
NFE: 60

* = VIOLATED CONSTRAINT(S)

TABLE XXXII

Test Results 47-Bar Planar Tower (IONED: 2,7)

ONE-DIMENSIONAL SEARCH: Golden Section

NDV = 44 NCON = 436 (Stress, Displacement, Buckling, and Frequency)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .53264E-01 Seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.60646E+04		.40012E+04		
DIRECT	NAC	0		10		
	NFE	756		1443		
1	OBJ		.59466E+04		.36788E+04	.35326E+04
SUMT	NAC		2		9	11
(EXT)	NFE		947		2154	2348
2						
SUMT	OBJ		.47985E+04		.30141E+04	.33997E+04
(LIN-	NAC		3		12	10
EXT-	NFE		2819		3192	2618
INT)						
3						
SUMT	OBJ		.53544E+04		.30141E+04	.33997E+04
(QUAD-	NAC		4		12	10
EXT-	NFE		2217		3192	2618
INT)						
4						
SUMT	OBJ		.53544E+04		.30141E+04	.33997E+04
(CUBIC	NAC		4		12	10
EXT-	NFE		2217		3192	2618
INT)						
5	OBJ		.40389E+04		.23645E+04	.23832E+04
SUMT	NAC		0*		9*	13*
(ALM)	NFE		711		7260	5434

CONMIN RESULTS: * = VIOLATED CONSTRAINT(S)

OBJ: .38078E+04
NAC: 9
NFE: 2021

TABLE XXXIII

Test Results 47-Bar Planar Tower (IONED: 3,8)

ONE-DIMENSIONAL SEARCH: Golden Section + Cubic Polynomial

NDV = 44 NCON = 436 (Stress, Displacement, Buckling, and Frequency)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .47939E-01 Seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0 DIRECT	OBJ	.61268E+04		.45382E+04		
	NAC	2		7		
	NFE	404		749		
1 SUMT (EXT)	OBJ		.55662E+04		.36696E+04	.36743E+04
	NAC		3		8	5
	NFE		1458		2390	2020
2 SUMT (LIN-EXT-INT)	OBJ		.46477E+04		.36015E+04	.35362E+04
	NAC		4		7	10
	NFE		2485		2451	2408
3 SUMT (QUAD-EXT-INT)	OBJ		.44886E+04		.36063E+04	.35387E+04
	NAC		5		7	8
	NFE		2438		2453	2247
4 SUMT (CUBIC EXT-INT)	OBJ		.45720E+04		.36102E+04	.35357E+04
	NAC		3		6	10
	NFE		2064		2282	2404
5 SUMT (ALM)	OBJ		.57343E+04		.32050E+04	.26490E+04
	NAC		2		5	5*
	NFE		883		3816	6258

CONMIN RESULTS:
OBJ: .38078E+04
NAC: 9
NFE: 2021

* = VIOLATED CONSTRAINT(S)

TABLE XXXIV

Test Results 47-Bar Planar Tower (IONED: 4,9)

ONE-DIMENSIONAL SEARCH: Bounds + Polynomial

NDV = 44 NCON = 436 (Stress, Displacement, Buckling, and Frequency)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .43639E-01 Seconds

OPTIMIZER: STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0 DIRECT	OBJ	.61115E+04		.44852E+04		
	NAC	2		9		
	NFE	523		970		
1 SUMT (EXT)	OBJ		.61499E+04		.50367E+04	.59280E+04
	NAC		2		4	2
	NFE		575		1101	726
2 SUMT (LIN-EXT-INT)	OBJ		.62486E+04		.57233E+04	.58948E+04
	NAC		2		0	2
	NFE		1214		1059	1174
3 SUMT (QUAD-EXT-INT)	OBJ		.62344E+04		.36405E+04	.53255E+04
	NAC		0		4	2
	NFE		1153		2651	1314
4 SUMT (CUBIC EXT-INT)	OBJ		.59038E+04		.58969E+04	.56291E+04
	NAC		2		2	2
	NFE		1741		1172	1226
5 SUMT (ALM)	OBJ		.61200E+04		.24724E+04	***
	NAC		4		4*	***
	NFE		871		3459	***

CONMIN RESULTS:
OBJ: .38078E+04
NAC: 9
NFE: 2021

* = VIOLATED CONSTRAINT(S)

TABLE XXXV

Test Results 234-Bar Space Tower (IONED: 2,7)

ONE-DIMENSIONAL SEARCH: Golden Section

NDV = 56 NCON = 3550 (Stress, Displacement, and Buckling)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .36011E+00 Seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0	OBJ	.75378E+05		.53526E+05		
DIRECT	NAC	1		2		
	NFE	297		1545		
1	OBJ		.48933E+05		.52357E+05	.46937E+05
SUMT	NAC		1*		0*	0*
(EXT)	NFE		2822		3166	1716
2						
SUMT	OBJ		.84527E+05		.84526E+05	.84525E+05
(LIN-	NAC		0*		0*	0*
EXT-	NFE		578		390	388
INT)						
3						
SUMT	OBJ		.84480E+05		.84480E+05	.84480E+05
(QUAD-	NAC		0*		0*	0*
EXT-	NFE		714		712	712
INT)						
4						
SUMT	OBJ		.84512E+05		.84512E+05	.84512E+05
(CUBIC	NAC		0*		0*	0*
EXT-	NFE		535		535	585
INT)						
5	OBJ		.46665E+05		.46849E+05	.43967E+05
SUMT	NAC		0*		2	2
(ALM)	NFE		4098		3949	5657

CONMIN RESULTS:
OBJ: .39353E+05
NAC: 4
NFE: 2946

* = VIOLATED CONSTRAINT(S)

TABLE XXXVI

Test Results 234-Bar Space Tower (IONED: 3,8)

ONE-DIMENSIONAL SEARCH: Golden Section + Cubic Polynomial

NDV = 56   NCON = 3550 (Stress, Displacement, and Buckling)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .34968E+00 Seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0 DIRECT	OBJ	.58577E+05		.54682E+05		
	NAC	1		2		
	NFE	2608		1343		
1 SUMT (EXT)	OBJ		.51327E+05		.53002E+05	.45573E+05
	NAC		2		2	0*
	NFE		3189		2917	3101
2 SUMT (LIN-EXT-INT)	OBJ		.84529E+05		.84528E+05	.84529E+05
	NAC		0*		0*	0*
	NFE		584		585	970
3 SUMT (QUAD-EXT-INT)	OBJ		.84480E+05		.84480E+05	.84480E+05
	NAC		0*		0*	0*
	NFE		715		713	713
4 SUMT (CUBIC EXT-INT)	OBJ		.84512E+05		.84512E+05	.84512E+05
	NAC		0*		0*	0*
	NFE		535		535	535
5 SUMT (ALM)	OBJ		.46507E+05		.45394E+05	.45364E+05
	NAC		2*		1	1
	NFE		4702		4100	4527

CONMIN RESULTS:

OBJ: .39353E+05

NAC: 4

NFE: 2946

* = VIOLATED CONSTRAINT(S)

TABLE XXXVII

Test Results 234-Bar Space Tower (IONED: 4,9)

ONE-DIMENSIONAL SEARCH: Bounds + Polynomial

NDV = 56 NCON = 3550 (Stress, Displacement, and Buckling)

AVERAGE CPU TIME PER FUNCTION EVALUATION: .32508E+00 Seconds

OPTIMIZER: / STRATEGY:		1 METHOD OF FEAS. DIR.	2 FLETCHER-REEVES	3 ROBUST M.F.D.	4 D.F.P.	5 B.F.G.S.
0 DIRECT	OBJ	.51504E+05		.54825E+05		
	NAC	2		1		
	NFE	2279		2070		
1 SUMT (EXT)	OBJ		.45275E+05		.46030E+05	.35709E+05
	NAC		0*		0*	1*
	NFE		1139		1971	1487
2 SUMT (LIN-EXT-INT)	OBJ		.84527E+05		.84526E+05	.84528E+05
	NAC		0*		0*	0*
	NFE		529		356	706
3 SUMT (QUAD-EXT-INT)	OBJ		.84480E+05		.84480E+05	.84480E+05
	NAC		0*		0*	0*
	NFE		698		698	698
4 SUMT (CUBIC EXT-INT)	OBJ		.84512E+05		.84512E+05	.84512E+05
	NAC		0*		0*	0*
	NFE		524		524	524
5 SUMT (ALM)	OBJ		.48394E+05		.45472E+05	.45321E+05
	NAC		3		1	2*
	NFE		3423		3769	3598

CONMIN RESULTS:
OBJ: .39353E+05
NAC: 4
NFE: 2946

* = VIOLATED CONSTRAINT(S)

LIST OF REFERENCES

1. Schmit, I.A., "Structural Design by Systematic Synthesis", Proceedings, Second Conference on Electronic Computation, ASCE, New York, 1960, pp. 105-122.

2. Schmit, L.A., "Structural Synthesis--Its Genesis and Development", AIAA Journal, Vol. 19, No. 10, Oct 1981, pp. 1249-1263.

3. Vanderplaats, G.N., "Structural Optimization Past, Present and Future", AIAA 1981 Annual Meeting and Technical Display, May 1981, Long Beach, California.

4. Vanderplaats, G.N., Sugimoto, H., and Sprague, C.M., "ADS-1: A New General-Purpose Optimization Algorithm" (to be published).

5. Vanderplaats, G.N., "CONMIN--A Fortran Program for Constrained Function Minimization, User's Manual", NASA TM X-62, 282, August 1973.

6. Madsen, L.E., and Vanderplaats, G.N., "COPES--A Fortran Congrol Program for Engineering Synthesis", NPS69-81-003, March 1982.

7. Fox, R.L., Optimization Methods for Engineering Design, Addison-Wesley, 1971.

8. Fiacco, A.V. and McCormick, G.P., Nonlinear Programming: Sequential Unconstrained Minimization Techniques, John Wiley and Sons, 1968.

9. Kavlie, D. and Moe, J., "Automated Design of Frame Structures", ASCE Journal of Structural Div., Vol. ST1, January 1971, pp. 33-62.

10. Cassis, J.H., Optimum Design of Structures Subject to Dynamic Loads, Ph.D. Thesis, University of California, Los Angeles, 1974.

11. Cassis, J.H. and Schmit, L.A., "On Implementation of the Extended Interior Penalty Function", International Journal of Numerical Methods in Engineering, Vol. 10, No. 1, 1976, pp. 3-23.

12. Haftka, R.T. and Starnes, J.H., Jr., "Applications of a Quadratic Extended Interior Penalty Function for Structural Optimization", AIAA Journal, Vol. 14, June 1976, pp. 718-724.

13. Prasad, B. and Haftka, R.T., "Optimum Structural Design with Plate Finite Elements", Journal of the Structural Division, ASCE, Vol. ST11, November 1979, pp. 2367-2382.

14. Prasad, B., "A Class of Generalized Variable Penalty Methods for Nonlinear Programming", Journal of Optimization Theory and Applications, Vol. 35, No. 2, October 1981, pp. 159-182.

15. Rockafellar, R.T., "The Multiplier Method of Hestenes and Powell Applied to Convex Programming", Journal of Optimization Theory and Application, Vol. 12, No. 6, 1973, pp. 555-562.

16. Pierre, D.A. and Lowe, M.J., "Mathematical Programming Via Augmented Lagrangians", Applied Mathematics and Computation Series, Addison-Wesley, 1975.

17. Powell, M.J.D., "Algorithms for Nonlinear Constraints that use Lagrangian Functions", Mathematical Programming, Vol. 14, No. 2, 1978, pp. 224-248.

18. Imai, K., Configuration Optimization of Trusses by the Multiplier Method, Ph.D. Thesis, University of California, Los Angeles, 1978.

19. Imai, K. and Schmit, L.A., "Configuration Optimization of Trusses", Journal of the Structural Division, ASCE, Vol. 107, No. ST5, May 1981, pp. 745-756.

20. Kelley, J.E., "The Cutting Plane Method for Solving Convex Programs", J. SIAM, 1960, pp. 703-712.

21. Moses, F., "Optimum Structural Design Using Linear Programming", Proc. ASCE, Vol. 90, ST6, 1964, pp. 89-104.

22. Baldur, R., "Structural Optimization by Inscribed Hyperspheres", Journal of Engineering Mechanics, ASCE, Vol. 98, No. EM3, June 1972, pp. 503-508.

23. Powell, M.J.D., "The Convergence of Variable Metric Methods for Nonlinearly Constrained Optimization Calculations", Proc. Non-Linear Programming Symposium 3, Madison, Wisconsin, 1978.

24. Powell, M.J.D., "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations", Report DAMTP77/NA2, University of Cambridge, England, 1977.

25. Zoutendijk, M., Methods of Feasible Directions, Elsevier Publishing Co., Amsterdam, 1960.

26. 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.

27. Fletcher, R. and Reeves, C.M., "Function Minimization by Conjugate Directions", Computer Journal, Vol. 7, No. 2, 1964, pp. 149-154.

28. Vanderplaats, G.N., "A Robust Feasible Directions Algorithm for Design Synthesis", Structural Dynamics and Materials Conference, Lake Tahoe, Nevada, May 2-4, 1983.

29. Davidon, W.C., "Variable Metric Method for Minimization", Argone National Laboratory, ANL-5990 Rev., University of Chicago, 1959.

30. Fletcher, R. and Powell, M.J.D., "A Rapidly Convergent Method for Minimization", Computer Journal, Vol. 6, No. 2, 1963, pp. 163-168.

31. Broydon, C.G., "The Convergence of a Class of Double Rank Minimization Algorithms", Parts I and II, J. Inst. Maths. Applns., Vol. 6, 1970, pp. 76-90 and 222-231.

32. Fletcher, R., "A New Approach to Variable Metric Algorithms", Computer Journal, Vol. 13, 1970, pp. 317-322.

33. Goldfarb, D., "A Family of Variable Metric Methods Derived by Variational Means", Maths. Comput., Vol. 24, 1970, pp. 23-36.

34. Shanno, D.F., "Conditioning of Quasi-Newton Methods for Function Minimization", Maths. Comput., Vol. 24, 1970, pp. 647-656.

35. Hadley, G., Non-Linear and Dynamic Programming, Addison-Wesley, 1964.

36. Fitzgerald, J.E., Development of a Computer Program for the Testing and Evaluation of Numerical Optimization Techniques, Master's Thesis, Naval Postgraduate School, Monterey, California, June 1982.

37. Dorn, W.S., Gomory, R.E., and Greenberg, H.J., "Automatic Design of Optimal Structures", Journal de Mechanique, Vol. 3, March 1964, pp. 25-52.

38. Colville, R.R., "A Comparative Study of Nonlinear Programming Codes", Report No. 320-2949, IBM Scientific Center, Philadelphia, Pennsylvania, June 1968.

39. Eason, E.D. and Fenton, R.G., "A Comparison of Numerical Optimization Methods for Engineering Design", Transactions of the ASME, Journal of Engineering for Industry, Vol. 96, No. 1, 1974, pp. 196-200.

40. Sangren, E. and Ragsdell, K.M., "The Utility of Nonlinear Programming Algorithms: A Comparative Study--Parts 1 and 2", Journal of Mechanical Design, ASME, July 1980, pp. 540-551.

41. Powell, M.J.D., "An Efficient Method for Finding the Minimum of a Function of Several Variables Without Calculating Derivatives", Computer Journal, Vol. 7, No. 4, 1964, pp. 303-307.

42. Vanderplaats, G.N., Numerical Optimization Techniques for Engineering Design: With Applications, McGraw-Hill (to be published).

43. Haftka, R.J., and Starnes, J.H., "Applications of a Quadratic Extended Interior Penalty Function for Structural Optimization", AIAA Journal, Vol. 14, No. 6, June 1976, pp. 718-724.

44. Prasad, B., "A Class of Generalized Variable Penalty Methods for Non-Linear Programming", Journal of Optimization Theory and Applications, Vol. 35, No. 2, October 1981, pp. 759-782.

45. Berke, L., and Khot, N.S., "Use of Optimality Criteria Methods for Large Scale Systems", AGARD Lecture Series No. 70 on Structural Optimization, AGARD-LS-70, 1974, pp. 1-29.

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

47. Jenkins, J.L., Application of Optimization Techniques to Naval Surface Combatant Ship Synthesis, Master's Thesis, Naval Postgraduate School, Monterey, California, October 1982.

48. Reed, M.B., Ship Synthesis Model for Naval Surface Ships, Master's Thesis, Massachusetts Institute of Technology, May 1976.

49. Johnson, D., and Brotton, D.M., "Optimum Elastic Design of Redundant Trusses", Discussion, J. Struct. Div., ASCE, Vol. 95, No. ST12, December 1969.

50. Vanderplaats, G.N., and Moses, F., "Automated Design of Trusses for Optimum Geometry", Journal of the Structural Division, ASCE, Vol. 98, No. ST3, March, 1972, pp. 671-690.

51. Felix, J. and Vanderplaats, G.N., "Configuration Optimization of Trusses Subject to Strength, Displacement and Frequency Constraints", Proc. Second ASME Computers in Engineering Conference, San Diego, August 1982.

52. Himmelblau, D.M., Applied Nonlinear Programming, McGraw-Hill, 1972.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Professor G.N. Vanderplaats, Code 69Vn Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	10
4.	Commandant (G-PTE-1) U.S. Coast Guard Headquarters 2100 Second Street Washington, D.C. 20593	2
5.	Commandant (G-ENE) U.S. Coast Guard Headquarters 2100 Second Street Washington, D.C. 20593	2
6.	Commanding Officer U.S. Coast Guard Research and Development Center Avery Point Groton, Connecticut 06340	2
7.	Dr. Hirokazu Miura NASA Ames Research Center MS 237-11 Moffett Field, California 94035	1
8.	Dr. Lucien Schmit Department of Mechanical & Structural Engineering 6731 Boelter Hall University of California Los Angeles Los Angeles, California 90024	1
9.	Dr. Jarek Sobieski NASA Langley Research Center MS 243 Hampton, Virginia 23665	1

10. Dr. Hiroyuki Sugimoto, Code 69 1
Department of Mechanical Engineering
Naval Postgraduate School
Monterey, California 93940

11. LCDR Chester M. Sprague 2
480 Norwich-New London Turnpike
Uncasville, Connecticut 06382

12. Chairman, Code 69 1
Department of Mechanical Engineering
Naval Postgraduate School
Monterey, California 93940

6-8

DTI