

EFFICIENT SINGLE-LEVEL SOLUTION OF HIERARCHICAL PROBLEMS IN
STRUCTURAL OPTIMIZATION

by

Rajiv R. Thareja

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in
Aerospace Engineering

APPROVED:

Raphael T. Haftka, Chairman

Henry J. Kelley

William L. Hallauer, Jr.

Eric R. Johnson

Rakesh K. Kapania

April, 1986

Blacksburg, Virginia

**EFFICIENT SINGLE-LEVEL SOLUTION OF HIERARCHICAL PROBLEMS IN
STRUCTURAL OPTIMIZATION**

by

Rajiv R. Thareja

Raphael T. Haftka, Chairman

Aerospace Engineering

(ABSTRACT)

Engineering design is hierarchical in nature, and if no attempt is made to benefit from this hierarchical nature, design optimization can be very expensive. There are two alternatives to taking advantage of the hierarchical nature of structural design problems. Multi-level optimization techniques incorporate the hierarchy at the formulation stage, and result in the coordinated optimization of a hierarchy of subsystems. The use of multi-level optimization techniques often necessitates the use of equality constraints. These constraints can sometimes cause numerical difficulties during optimization. Single-level decomposition techniques take advantage of the hierarchical nature to reduce the optimization cost.

In this research the decomposition approach has been followed to reduce the computational effort in a single-level design space. A decoupling technique has been developed that retains the advantages of a partitioned system of smaller independent subsystems without an increase in the total number of design variables. A penalty function formulation using Newton's method for the solution of a sequence of unconstrained minimizations was

11/10/86
MCR

employed. The optimization of the decoupled system is cheaper due to (i) cheaper evaluation of the hessian matrix by taking advantage of its sparsity, (ii) fewer global analyses for constraint derivative calculations, and (iii) utilizing the decoupled nature of the hessian matrix in the solution process. Further, the memory requirements of the decoupled system are much less than that of the original coupled system. These benefits increase substantially for design problems with larger and larger number of detailed design variables.

Orthotropic material properties as stiffness global variables have been shown to be effective as global variables for panels in a hierarchical wing design formulation.

The proposed decoupling technique was implemented to minimize the volume of a portal frame and a wing box. Computational savings of up to 50 percent have been obtained for medium sized problems. The savings increase as the size of the problem and the amount of decoupling is increased. The procedure is simple to implement. For truly large systems this decoupling technique provides the necessary reduction of computational effort to make the optimization process viable.

Acknowledgements

The author wishes to express his sincere gratitude to Professor Raphael T. Haftka, without whose constant guidance, encouragement and support this research would not have been accomplished. His broad-based knowledge has been a source of inspiration.

The author is also indebted to Professor Henry J. Kelley, Professor William L. Hallauer, Jr., Professor Eric R. Johnson and Professor Rakesh K. Kapania for their helpful suggestions and comments throughout this effort, and for serving on his committee.

The support of Dr. James H. Starnes, Jr. is gratefully acknowledged.

This research was sponsored by a research grant from National Aeronautics and Space Administration (NASA) through grant NAG-1-168.

The author would also like to thank _____ for her constant support and encouragement and for just being there.

Table of Contents

1.0 Introduction	1
1.1 Hierarchical Nature of Structural Design	1
1.2 Multi-Level Structural Design	3
1.3 Decomposition During Solution	8
1.4 Objectives of the Present Work	9
1.5 Outline	10
2.0 Penalty Function Solution Procedure	12
2.1 Newton's Method with Approximate Second Derivatives	16
2.2 Constraint Approximations	20
3.0 Formulation and Benefits of Hierarchical Problem	24
3.1 Formulation of the Hierarchical Problem	24
3.2 Penalty Function Equivalent of Coupled System	28
3.3 Penalty Function Equivalent of Decoupled System	28
3.4 Benefits of Decoupling in the Solution Process	31
3.4.1 Savings in Evaluation of Hessian Matrix	32

3.4.2	Savings in Constraint Gradients	32
3.4.3	Savings in Solution of Direction Equation	33
4.0	Solution Techniques for Single-Level and Multi-Level Formulations	35
4.1	Single-Level Formulation	39
4.2	Single-Level Solution with Equality Constraints	40
4.3	Multi-Level Solution	41
4.4	Description of Frame Example	42
4.5	Optimization Techniques for the Frame Example	45
4.6	Results for the Frame Example	46
4.7	Conclusions for the Frame Example	58
5.0	Solution using Decoupling Intermediate Variables	60
5.1	Derivative Transformation	63
5.1.1	Single Subsystem Transformation Matrix	68
5.1.2	Multi-Subsystem Equations	70
5.2	Choosing the Eliminated Set	72
5.3	Calculating a Search Direction	73
5.4	Calculation of Derivatives of Constraints	76
5.4.1	Perturbing Elements of R	77
5.4.2	Perturbing Elements of D	78
5.4.3	Computing Constraint Derivatives with Respect to Original Variables	79
6.0	Numerical Results for Frame Example	81
6.1	Numerical Results for Portal Frame FRAME015	85
6.2	Numerical Results for Portal Frame FRAME050	93
6.3	Numerical Results for Portal Frame FRAME125	108

6.4	Comparison of Results for All Models of the Portal Frame	117
7.0	Numerical Results for Wing Box	126
7.1	Development of Global Variables for Wing Panels	126
7.2	Description of Wing Box	132
7.3	Numerical Results for Wing Box	134
8.0	Conclusions and Recommendations	144
	Appendix A. Hessian Matrix in Decoupled Space	147
	References	154
	Vita	162

List of Illustrations

Figure 1.	Geometry of Portal Frame with I-Beam Cross Section	26
Figure 2.	Geometry of Portal Frame with Hat-Stiffener Cross Section	37
Figure 3.	Optimum Volume as a Function of the Cross Sectional Area of Beam 2 for all Optimization Techniques	57
Figure 4.	Finite Element Model for Portal Frame FRAME015	82
Figure 5.	Optimization Convergence History for Various Approximations for Portal Frame FRAME015	86
Figure 6.	Optimization Convergence History for Coupled and Decoupled Systems with Exact Constraints for Portal Frame FRAME015	94
Figure 7.	CPU Times for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME015	95
Figure 8.	Number of Global Analyses for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME015	96
Figure 9.	Finite Element Model for Portal Frame FRAME50	97
Figure 10.	Optimization Convergence History for Various Approximations for Portal Frame FRAME050	98
Figure 11.	Optimization Convergence History for Coupled and Decoupled Systems with Exact Constraints for Portal Frame FRAME050	104
Figure 12.	CPU Times for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME050	105

Figure 13. Number of Global Analyses for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME050	106
Figure 14. Finite Element Model for Portal Frame FRAME125	109
Figure 15. Optimization Convergence History for Exact and Linear Approximation for Constraints for Portal Frame FRAME125	110
Figure 16. Optimization Convergence History for Coupled and Decoupled Systems with Exact Constraints for Portal Frame FRAME125	113
Figure 17. CPU Times for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME125 ..	114
Figure 18. Number of Global Analyses for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME125	115
Figure 19. Cross Sectional Area Variation for Beam 1 for all Models of the Portal Frame	119
Figure 20. Cross Sectional Area Variation for Beam 2 for all Models of the Portal Frame	120
Figure 21. Total CPU Times for Coupled and Decoupled Systems for all Models of the Portal Frame	122
Figure 22. Hessian Setup Time for Coupled and Decoupled Systems for all Models of the Portal Frame	123
Figure 23. Constraint Gradient Time for Coupled and Decoupled Systems for all Models of the Portal Frame	124
Figure 24. Number of Global Analyses for Coupled and Decoupled Systems for all Models of the Portal Frame	125
Figure 25. Finite Element Model for Wing Box	127
Figure 26. Details of Hat-Stiffeners for the Cover Panels	129
Figure 27. Optimization Convergence History for Various Approximations for Wing Box	139
Figure 28. Optimization Convergence History for Coupled and Decoupled Systems with Exact Constraints for Wing Box	141

Figure 29. CPU Times for Coupled and Decoupled Systems with Various Approximations for Wing Box 142

Figure 30. Number of Global Analyses for Coupled and Decoupled Systems with Various Approximations for Wing Box 143

List of Tables

Table 1.	Portal Frame Example - Design Information	43
Table 2.	Optimization Results for Portal Frame Example, NEWSUMT-A Program	47
Table 3.	Optimization Results for Portal Frame Example, ADS Program, Method of Feasible Directions	49
Table 4.	Optimization Results for Portal Frame Example, ADS Program, Modified Method of Feasible Directions	50
Table 5.	Optimization Results for Portal Frame Example, ADS Program, SUMT - Exterior Penalty Function	51
Table 6.	Optimization Results for Portal Frame Example, ADS Program, SUMT - Linear Extended Interior Penalty Function	53
Table 7.	Optimization Results for Portal Frame Example, ADS Program, SUMT - Quadratic Extended Interior Penalty Function	54
Table 8.	Optimization Results for Portal Frame Example, ADS Program, SUMT - Linear Extended Interior Penalty Function	55
Table 9.	Optimization Results for Portal Frame Example, ADS Program, Augmented Lagrange Multiplier	56
Table 10.	Portal Frame Example - Design Information	84
Table 11.	Optimization Results for Portal Frame FRAME015 using Exact Constraints	88
Table 12.	Optimization Results for Portal Frame FRAME015 using Linear Approximation for Constraints	89

Table 13. Optimization Results for Portal Frame FRAME015 using Quadratic Approximation for Constraints	90
Table 14. Optimization Results for Portal Frame FRAME015 using Reciprocal Approximation for Constraints	91
Table 15. Optimization Results for Portal Frame FRAME015 using Conservative Approximation for Constraints	92
Table 16. Optimization Results for Portal Frame FRAME050 using Exact Constraints	99
Table 17. Optimization Results for Portal Frame FRAME050 using Linear Approximation for Constraints	100
Table 18. Optimization Results for Portal Frame FRAME050 using Quadratic Approximation for Constraints	101
Table 19. Optimization Results for Portal Frame FRAME050 using Reciprocal Approximation for Constraints	102
Table 20. Optimization Results for Portal Frame FRAME050 using Conservative Approximation for Constraints	103
Table 21. Optimization Results for Portal Frame FRAME125 using Exact Constraints	111
Table 22. Optimization Results for Portal Frame FRAME125 using Linear Approximation for Constraints	112
Table 23. Comparison of Results for all Models of the Portal Frame ...	118
Table 24. Wings Box Example - Design Information	133
Table 25. Wings Box Example - Design Variable Information	135
Table 26. Optimization Results for Wing Box using Exact Constraints ..	136
Table 27. Optimization Results for Wing Box using Linear Approximation for Constraints	137
Table 28. Optimization Results for Wing Box using Quadratic Approximation for Constraints	138

1.0 Introduction

1.1 *Hierarchical Nature of Structural Design*

The design of complex structures typically proceeds at several levels of detail. The complete structure is usually far too complicated to be analyzed in great detail. At the same time a detailed analysis of some aspects of the design must be performed to achieve a superior design.

As an example consider the current practice for the design of an aircraft wing. The overall design of the wing proceeds on the basis of a beam analysis or a gross finite element model. From this analysis the loads acting on the major components of the wing such as individual panels are obtained. Each panel is then designed based on the assumption that these loads are fixed. The panel design is then followed by detail design to prevent stress concentration around cutouts and discontinuities. Again it is assumed that the

loads from the panel design do not change when the details of cutouts are changed. This approach neglects the effects of the redesign of one part of the structure on other parts, and does not lend itself to exploit the interaction between local structures. As a result a truly optimized structure cannot be achieved since there is no mechanism for redistribution of forces or material between local structures.

One way to solve the problem of exploiting the interaction between local structures is to carry out a local-global design simultaneously, i.e., to consider all the design variables and constraints at the same time. This usually involves hundreds or even thousands of design variables and constraints. The design task becomes very expensive and optimization of such a system can be prohibitive with respect to computational effort and memory requirements.

Many engineering systems have the property that their subsystems are only weakly coupled. This is especially true in wing design, where changes in the detail design in one panel of the wing have only a small effect on the forces between all panels, and, subsequently, the response of other remotely located panels. Even though this effect of load redistribution is weak, it cannot be neglected in the analysis and optimization process. Because the detail design of one part of the structure has only a weak effect on the detail design of another part, the problem naturally calls for the use of multi-level optimization techniques. This is true for many engineering and economic systems, and, as a result, there has been great interest in such techniques in the past few years (Refs 1 - 40).

1.2 Multi-Level Structural Design

One of the key problems in solving the weakly coupled problem of global local design is to formulate it as a hierarchical problem where the lower-level design problems are truly decoupled.

There are several techniques (Refs. 1 - 8) to convert the structural design problem into a two-level problem of global and detailed design. One approach that imitates current design practice (e.g. Ref 2 and 3) is to divide the complete structure into substructures and assume that the inter-substructure forces are fixed for detail design and are dictated only by the global level design. This substructuring approach completely neglects the interaction between substructures and so it does not lend itself well to handling of global constraints such as overall stiffness requirements.

Kirsch discusses two general approaches to multi-level formulation for optimization (Ref 13). In the model coordination method the design variables are divided into interaction or coordinating variables between the subsystems and subsystem variables. At the lower level the subsystems are optimized for fixed values of the interaction variables, while the interaction variables are optimized at the upper level. This process is repeated until convergence is achieved. In the goal coordination approach, the subsystems are uncoupled completely with interconnection variables defined between subsystems. The lower level optimization modifies the value of the subsystem variables, while

the upper level optimization ensures consistency through equality constraints on interconnection variables.

Schmit and coworkers (Refs 28 - 31) tackled the problem of decoupling the local problems by having the local level objective to be minimized as the change in local stiffness from its global value, while satisfying side constraints and local buckling constraints. The global level variables were the stiffnesses while the objective was to minimize the structural weight satisfying global constraints on strength, stresses, displacements and global buckling. By using stiffness variables as global design variables, the local design problems were decoupled because for fixed global stiffness the loads on local structures are fixed. This two-level approach along with the use of the global variables at the higher level provided the mechanism of permitting changes in global level quantities while allowing each lower-level problem to be optimized separately. These global variables are the mechanism by which a given large system is decoupled into several smaller subsystems.

Once global variables have been identified for a given problem, reduction in the number of global analyses required is possible if the number of local variables is larger than the number of global variables. Constraint derivatives with respect to the global variables require a global analysis. On the other hand constraint derivatives with respect to the local variables can be obtained cheaply by using the former derivatives and the chain rule of differentiation.

Researchers have developed two different techniques to capitalize on the benefits ensuing from the decoupling. One technique is to break up the larger

problem into several smaller problems at the formulation stage itself and use multi-level techniques to solve the resulting hierarchical optimization problem. The other is to use a single-level optimization technique, but to take advantage of the hierarchical nature of the system in the solution stage of the optimization.

Most of the research effort in multi-level structural optimization has been motivated by the fact that for very large structural problems it has been impossible to perform a single-level optimization due to large memory requirements.

For large systems multi-level decomposition methods have been developed to break up a large system into several smaller subsystems. Each subsystem is then independently optimized. At a higher level coordination between the sublevels is used to integrate the optimization of the overall system (see Refs. 11 and 12). By decomposing a large system it is possible to handle problems that could not be solved before. For a detailed historical development of formal and intuitive decomposition methods see Ref 42. A multi-level design technique typically is more complicated to implement and usually more expensive, but it can be handled with limited memories and utilize parallel computing capabilities (Ref 32).

With the idea of formalizing a multi-level optimization approach for general engineering systems Sobieski suggested a method for decomposing large optimization problems into a number of smaller subproblems (Ref 32). A tree hierarchy was used to achieve such a decomposition. A cumulative constraint

violation function was minimized for each subsystem by modifying its local design variables while holding constant the higher level variables. The sensitivity derivatives of these minima with respect to the higher level variables were also computed. At the highest level linear extrapolations of the lower level minima were generated, and the system was optimized for an objective function augmented by the cumulative constraint violation function from the lower levels subjected to the highest level constraints. This process was repeated going from higher to lower levels until convergence was achieved. Sobieski and coworkers (Refs 32 - 34) used this approach to minimize the weight of a portal frame subject to stress, displacement and buckling constraints. They devised a two-level (Ref 33) and a three-level (Ref 35) hierarchical structure for the frame model by considering greater and greater levels of details.

In a typical multi-level optimization approach lower level subsystems may need to be optimized many times, and this can make the approach much more expensive than a single-level optimization. To avoid the repeated optimization of lower-level subsystems Sobieski and coworkers used a linear extrapolation of the lower-level optima based on their sensitivity derivatives.

The derivatives of the lower level optima, known as sensitivity derivatives have received considerable attention recently (Ref 41 - 47). Unfortunately, optimum sensitivity derivatives can be discontinuous functions of the higher level design variables, thereby creating numerical difficulties for the higher

level optimizer (Ref 17 and 41). Additionally, for some choices of the global design variables these lower level optima may not exist at all.

Often in order to coordinate the higher level and lower levels and to assure consistency between lower and upper level variables there is need to impose equality constraints. These equality constraints may be considered as a device to transform a single-level problem into a multi-level one. For simple problems, it may be possible to use the equality constraints to eliminate some of the local design variables and replace them by global variables. For example, in the portal frame problem considered by Sobieski et. al. (Refs 33 and 35) the equality constraints were implicitly enforced by eliminating variables at the lower level. For more complicated equality constraints this may not be possible, and the equality constraints may have to be retained during optimization. However, equality constraints are more difficult to handle than inequality constraints and as a result the multi-level optimization problem may be more ill-conditioned than the single-level problem.

Because of the difficulties associated with multi-level optimization, i.e., discontinuous optimum sensitivity derivatives, equality constraints and the nonexistence of the subsystem optima, other alternatives for solving large weakly coupled problems have been considered. These typically involve a single-level formulation but use a decomposition technique for the solution of the single-level problem.

1.3 Decomposition During Solution

For linear problems linear programming (LP) decomposition methods are used to convert a large problem into a sequence of smaller LP problems. The Dantzig-Wolfe decomposition method (Ref 9) was the first to take advantage in the solution process of a block-diagonal structure that exists in most large linear problems. Rosen (Ref 10) suggested a modified approach that had better convergence properties. In his approach two problems are solved alternately until the optimal solution is obtained. These algorithms are guaranteed to converge to the correct optimum only for linear or convex problems.

Recently Haftka (Ref 17) proposed a method for a penalty function solution of hierarchical nonlinear programming problems. Newton's method was used to solve a sequence of unconstrained problems and it was shown that a hierarchical problem results in a block sparse hessian (second derivative) matrix. Significant savings in computational times were achieved in the assembly of the hessian matrix and solution of the direction vector. Additionally, the sparsity of the hessian matrix allows the solution of large problems without excessive memory requirements. The solution procedure was still single-level, so that the technique preserved most of the advantages of both the single-level and multi-level techniques of optimization.

Haftka's approach has been applied to medium sized problems. Global variables are used as intermediate variables in the solution process. The

linear relationship between global and local variable increments is used to eliminate local variables in the equations for derivatives and search directions.

1.4 Objectives of the Present Work

The objectives of the present research are to:

1. Develop a hierarchical wing design formulation by introducing stiffness global variables.
2. Check the effect of using equality constraints on the optimization solution process as the first objective requires the use of equality constraints for the multi-level formulation.
3. Develop a single-level solution method based on an extension of the decomposition approach of Ref 17, and the use of global variables as intervening variables.

1.5 Outline

Chapter 2 discusses the penalty function solution procedure that was used to solve the optimization problem. Modifications made to an existing nonlinear optimization package called NEWSUMT that uses an extended interior penalty function with Newton's method for solution of unconstrained problems are also discussed. The major changes are the addition of the capability to handle equality constraints, and the use of more extensive constraint approximation techniques to reduce the number of analyses required to compute constraint derivatives.

The description of the hierarchical formulation and the benefits it offers in the solution of the optimization of large systems is discussed in Chapter 3.

Chapter 4 examines the impact of the equality constraints on the optimization of a portal frame example. In this chapter a single-level formulation with a single-level solution, a two-level formulation with a single-level solution, and a two-level formulation with a two-level solution for the frame problem are presented. It is shown that the equality constraints sometimes lead to numerical problems.

The proposed technique of using global variables as intervening variables and using a decomposition approach in a single-level design space is discussed in detail in Chapter 5. The details of computing the constraint derivatives while the local and global variables are perturbed are also discussed in this chapter.

Numerical results from applying the proposed technique to the portal frame problem are presented in Chapter 6. Results are compared with those obtained from a single-level optimization procedure, and it is shown that similar designs are obtained with significant savings in computational effort.

Chapter 7 presents the hierarchical formulation of the wing design problem and the results obtained by applying the proposed decomposition technique to this problem.

Some conclusions about this research effort and recommendations for further research are presented in Chapter 8.

2.0 Penalty Function Solution Procedure

The use of mathematical programming techniques in engineering design has increased rapidly in the past two decades. At the beginning of that period special purpose optimization algorithms were employed with special purpose analysis programs. The increasing level of maturity in this discipline is reflected in the emergence of general purpose software packages designed for a wide range of applications (e.g. Ref 48, 50, 51 - 53).

However, even though thousands of papers have been published on the use of optimization algorithms to engineering design, the number of applications to the design of real vehicles remains very small (e.g. Ref 54). One of the main reasons for this state of affairs is the high cost of employing optimization techniques in the design process. An optimization algorithm may require hundreds or thousands of evaluations of the objective function and constraints. In many engineering applications a single evaluation of the constraints or the objective function may be very costly in itself. A typical

example is the design of a complex structure subject to stress and displacement constraints. A costly finite element analysis is usually required for the evaluation of the constraints. It is no wonder that the use of optimization techniques for such problems may be prohibitively expensive.

A solution to the cost problem inherent in the use of optimization methods was suggested by Schmit and coworkers (Ref 55) in the early seventies. It consists of employing constraint approximations to reduce the number of times that these constraints have to be calculated exactly. In the years that followed several papers have presented various constraint approximation techniques (e.g. Ref 56 - 62). The popularity of the ideas was underscored in a recent international symposium on Structural optimization (Ref 63), where a large number of papers (e.g. Ref 64 - 70) employed approximation techniques. One of the more popular approximation techniques has been the reciprocal constraint approximation which is a linear Taylor series expansion in the reciprocal of the design variables.

The use of constraint approximations sometimes requires a strategy for selecting move-limits that keep the design in a region where the approximation is valid. Additionally, a strategy for switching between the exact and approximate analyses is also required. Currently, optimization software packages include some of the move-limit and switching strategies. However, in most cases it is the user who has to provide an interface to his analysis program which controls the move-limits and the switching (Ref 71 is an

example of a design system composed of a general purpose optimizer, an analysis program and the interface).

In this chapter we shall consider the optimization of a general nonlinear objective function subject to a set of nonlinear equality and inequality constraints.

Consider the general optimization problem of the form

$$\text{minimize } f(X_1, X_2, \dots, X_{NDV})$$

$$\text{such that } g_i(X_1, X_2, \dots, X_{NDV}) \geq 0, \quad i = 1, \dots, n_i \quad [2.1]$$

$$\text{and } h_i(X_1, X_2, \dots, X_{NDV}) = 0, \quad i = 1, \dots, n_e$$

Among the numerous numerical techniques available to obtain a solution to equation [2.1] one popular technique is the SUMT approach. In this technique the constrained optimization problem is transformed into a series of unconstrained minimization problems and solved by the Sequence of Unconstrained Minimization Technique (Ref 72) using the quadratic extended interior penalty function method (Ref 49). The NEWSUMT optimization package (Ref 48) was modified to handle equality constraints (Ref 76 and 77) using exterior penalty terms. The penalty from each equality constraint is proportional to the square of the violation. The form of the augmented function φ , is

$$\varphi(X, r) = f(X) + r \sum_{i=1}^{n_i} p(g_i) + \frac{\beta}{\sqrt{r}} \sum_{i=1}^{n_e} q(h_i) \quad [2.2]$$

where

\mathbf{X} = design variable vector of length NDV

r = penalty multiplier

n_i = number of inequality constraints

g = inequality constraint

β = multiplier used to adjust equality constraint penalty

n_e = number of equality constraints

h = equality constraint

The penalty $p(g_i)$ associated with the i -th inequality constraint is defined as:

$$p(g_i) = \begin{cases} \frac{1}{g_i} & \text{if } g_i \geq g_0 \\ \frac{1}{g_0} \left[\left(\frac{g_i}{g_0}\right)^2 - 3\left(\frac{g_i}{g_0}\right) + 3 \right] & \text{if } g_i \leq g_0 \end{cases} \quad [2.3]$$

$$g_0 = C \sqrt{r}$$

where g_0 is a transition parameter and C is a constant. The function $p(g_i)$ is thus defined as an interior penalty function in most of the feasible design domain. It is defined as a quadratic exterior penalty function in a small part of the feasible domain ($g_i \leq g_0$) and in the infeasible domain. The penalty function is continuous up to its second derivatives.

Also, to ensure a positive value for a violation the penalty $q(h_i)$ associated with the i -th equality constraint is defined as an exterior penalty function of the form

$$q(h_i) = h_i^2 \quad [2.4]$$

The solution of the optimization problem is obtained by minimizing the function φ for a decreasing sequence of r values, using Newton's method with approximate second derivatives of the penalty terms (Ref 73). The optimization is started with the penalty multiplier r at a value R_{init} and the total function φ given by equation [2.2] is minimized. This is called a response surface. Then r is reduced by multiplying it by a factor R_{mult} and another response surface is computed. This process is continued until convergence to a desired accuracy is achieved.

2.1 Newton's Method with Approximate Second Derivatives

The direction vector \mathbf{S} , that minimizes $\varphi(\mathbf{X}, r)$ is found by using Newton's method with a one-dimensional search.

$$\mathbf{S} = \Delta\mathbf{X} = \mathbf{X} - \mathbf{X}_0 = -\alpha \mathbf{H}^{-1} \mathbf{G} \quad [2.5]$$

where

α is the step size obtained by a one dimensional search.

\mathbf{S} is the direction vector.

\mathbf{G} is the gradient vector of φ .

\mathbf{X}_0 is the initial design variable vector.

\mathbf{H} is the hessian matrix (second derivative matrix of φ).

$$G_i = \frac{\partial \varphi}{\partial x_i}$$

$$= \frac{\partial f}{\partial x_i} + r \sum_{k=1}^{n_i} \frac{\partial p(g_k)}{\partial x_i} + \frac{\beta}{\sqrt{r}} \sum_{k=1}^{n_e} \frac{\partial q(h_k)}{\partial x_i} \quad [2.6]$$

$$H_{ij} = \frac{\partial^2 \varphi}{\partial x_i \partial x_j}$$

$$= \frac{\partial^2 f}{\partial x_i \partial x_j} + r \sum_{k=1}^{n_i} \frac{\partial^2 p(g_k)}{\partial x_i \partial x_j} + \frac{\beta}{\sqrt{r}} \sum_{k=1}^{n_e} \frac{\partial^2 q(h_k)}{\partial x_i \partial x_j} \quad [2.7]$$

For the contribution from the inequality constraints we have the following two relations for the gradient vector \mathbf{G} and the hessian matrix \mathbf{H} .

$$\frac{\partial p}{\partial x_i} = \begin{cases} \frac{\partial g_k}{\partial x_i} \left[2 \frac{g_k}{g_0} - 3 \right] & \text{if } g_k \leq g_0 \\ - \frac{1}{g_k^2} \frac{\partial g_k}{\partial x_i} & \text{if } g_k \geq g_0 \end{cases} \quad [2.8]$$

$$\frac{\partial^2 p(g_k)}{\partial x_i \partial x_j} = \begin{cases} \frac{1}{g_k^3} \left[2 \frac{\partial g_k}{\partial x_i} \frac{\partial g_k}{\partial x_j} - g_k \frac{\partial^2 g_k}{\partial x_i \partial x_j} \right] & \text{if } g_k \geq g_0 \\ \frac{1}{g_0^3} \left[2 \frac{\partial g_k}{\partial x_i} \frac{\partial g_k}{\partial x_j} + g_0 \left(2 \frac{g_k}{g_0} - 3 \right) \frac{\partial^2 g_k}{\partial x_i \partial x_j} \right] & \text{if } g_k \leq g_0 \end{cases} \quad [2.9]$$

For equality constraints we have the following two relations for the contributions to the gradient vector \mathbf{G} and the hessian matrix \mathbf{H} .

$$\frac{\partial q(h_k)}{\partial x_i} = 2 h_k \frac{\partial h_k}{\partial x_i} \quad [2.10]$$

$$\frac{\partial^2 q(h_k)}{\partial x_i \partial x_j} = 2 \frac{\partial h_k}{\partial x_i} \frac{\partial h_k}{\partial x_j} + 2 h_k \frac{\partial^2 h_k}{\partial x_i \partial x_j} \quad [2.11]$$

Approximate second derivatives are evaluated as suggested by Haftka (Ref 73). For critical constraints, the second term in equation [2.9] is very small compared to the first term and for noncritical constraints $g_k(\mathbf{X})$ is large and the entire second derivative can be neglected. Therefore equation [2.9] becomes

$$\frac{\partial^2 p(g_k)}{\partial x_i \partial x_j} = \begin{cases} \frac{2}{g_k^3} \frac{\partial g_k}{\partial x_i} \frac{\partial g_k}{\partial x_j} & \text{if } g_k \geq g_0 \\ \frac{2}{g_0^3} \frac{\partial g_k}{\partial x_i} \frac{\partial g_k}{\partial x_j} & \text{if } g_k \leq g_0 \end{cases} \quad [2.12]$$

Also, the second term in equation [2.11] goes to zero as the optimization progresses and the equality is satisfied. Thus equation [2.11] becomes

$$\frac{\partial^2 q(h_k)}{\partial x_i \partial x_j} = 2 \frac{\partial h_k}{\partial x_i} \frac{\partial h_k}{\partial x_j} \quad [2.13]$$

These approximations for the second derivatives of φ require only the first derivatives of the constraints, therefore the computational effort required in finding the search direction in a one dimensional minimization is reduced. These approximate second derivatives are used in evaluating the hessian matrix and thus finding the search direction. One dimensional minimizations to find α in NEWSUMT-A are carried out by first trapping an initial interval of uncertainty and then applying the golden section method for the precise value.

We denote the total number of design variables as NDV and the total number of constraints as $NCON$ where $NCON = n_i + n_e$ in equations [2.1] and [2.2].

From the above relations it is evident that for the gradient vector we need to evaluate $NDV \times NCON$ terms, while we need $NCON \times NDV \times NDV / 2$

terms for the hessian matrix, by noting that the hessian matrix is symmetric and therefore only the upper diagonal needs to be evaluated and stored. The evaluation of the hessian matrix can constitute a major computational effort for large values of the number of design variables and constraints.

Though the hessian matrix \mathbf{H} in equation [2.5] is exact, in practice it is approximated from the first derivatives of the constraints with respect to the design variables. Also, errors caused by inaccuracies in the derivatives usually computed by finite-differences result in errors in the terms of the hessian matrix. In reality then instead of equation [2.5] the equation that is solved is

$$\mathbf{s} = \Delta\mathbf{X} = \mathbf{X} - \mathbf{X}_0 = -\alpha [\mathbf{H} + \mathbf{H}_c]^{-1} \mathbf{G} \quad [2.14]$$

where \mathbf{H}_c is a diagonal matrix with terms $\varepsilon \mathbf{H}_{ii}$ along the diagonal. These correction terms are used to make the original hessian matrix \mathbf{H} more diagonally dominant to prevent a singular matrix. Typical values of ε are in the range 0.01 - 0.05.

2.2 Constraint Approximations

The calculation of the constraints which are usually nonlinear implicit functions of the design variables, is often expensive. One way to alleviate this problem is to use constraint approximations based on Taylor series

expansions about a design point X_0 . When constraint approximations are used, it may be necessary to impose move-limits which prevent the design from moving far from X_0 , where the approximations could be inaccurate. A scheme to do so may be employed for further reducing the computational cost (Ref 77).

In NEWSUMT-A first order Taylor series expansions are used for constructing global approximate constraint functions. The constraint functions and their derivatives with respect to the design variables are computed using an exact analysis. Approximate constraints are then used to improve the design without performing any more exact analyses. The user has the option of controlling the frequency of updating the approximation. It is possible to specify that a single approximation be used throughout the entire optimization process, or that it be updated several times within a single unconstrained minimization. Additionally, different (local) constraint approximations are used during one-dimensional searches. The approximations employed during one dimensional searches are based on the values of the constraints and their derivatives at the initial point of the one dimensional search (which could be exact or obtained by the global approximation).

In discussing the various forms of constraint approximation used in the NEWSUMT-A optimizer, approximations which are based on the function and its derivatives at a single point X_0 are described first.

The simplest form of constraint approximation is a linear function of the design variables using a first order Taylor series expansion.

$$g_D(\mathbf{X}) = g(\mathbf{X}_0) + \sum_{i=1}^{NDV} (x_i - x_{0i}) \frac{\partial g(\mathbf{X}_0)}{\partial x_i} \quad [2.15]$$

Another form of approximation that is often used is a linear function of the reciprocals of the design variables (Ref 55). This reciprocal approximation is exact for stresses and displacements of a statically determinate structure. It is more accurate for stresses and displacements than the linear approximation for a statically indeterminate structure (Ref 57).

$$g_R(\mathbf{X}) = g(\mathbf{X}_0) + \sum_{i=1}^{NDV} (x_i - x_{0i}) \frac{x_{0i}}{x_i} \frac{\partial g(\mathbf{X}_0)}{\partial x_i} \quad [2.16]$$

Finally, another first order approximation is a conservative approximation (Ref 60), which assumes it is better to be more conservative than to strive for maximum accuracy. With this approximation, a given constraint may have a linear approximation with respect to one design variable and a reciprocal approximation with respect to another design variable.

$$g_C(\mathbf{X}) = g(\mathbf{X}_0) + \sum_{i=1}^{NDV} B_i (x_i - x_{0i}) \frac{\partial g(\mathbf{X}_0)}{\partial x_i} \quad [2.17]$$

where

$$B_i = \begin{cases} 1 & \text{if } x_i \frac{\partial g}{\partial x_i} < 0 \\ \frac{x_{oi}}{x_i} & \text{if } x_i \frac{\partial g}{\partial x_i} > 0 \end{cases}$$

In some cases where the first order approximations are not sufficient, approximations based on second order Taylor series expansions are used to get more accurate results. However these quadratic approximations require substantial computational effort and storage, since the approximation is based on second derivatives of constraints with respect to design variables.

$$g_Q(\mathbf{X}) = g(\mathbf{X}_0) + \sum_{i=1}^{NDV} (x_i - x_{oi}) \frac{\partial g(\mathbf{X}_0)}{\partial x_i} + \frac{1}{2} \sum_{i=1}^{NDV} \sum_{j=1}^{NDV} (x_i - x_{oi}) \frac{\partial^2 g(\mathbf{X}_0)}{\partial x_i \partial x_j} (x_j - x_{oj}) \quad [2.18]$$

The user can use any one of the first order approximations as global approximations and both first and second order approximations as local approximations for the one dimensional minimizations.

3.0 Formulation and Benefits of Hierarchical Problem

3.1 Formulation of the Hierarchical Problem

In Chapter 1 we discussed some of the characteristics of large structural systems. Let us now develop the expressions for describing the hierarchical nature of typical structural systems. It is assumed that the system can be divided into s substructures each with its own independent design variable vector X_i defined for $i = 1, \dots, s$.

Consider a structural optimization problem of the form

$$\text{minimize } f(X_1, X_2, \dots, X_s) \quad [3.1]$$

$$\text{such that } g_j(X_1, X_2, \dots, X_s) \geq 0, \quad i = 1, \dots, m$$

where X_i is a vector of local design variables associated with the i -th part of the structure. In the form [3.1] the system is fully coupled.

Often it is possible to find a vector of decoupling (or global) variables Y such that the system [3.1] may be written as

$$\text{minimize } f = f_o(Y) + \sum_{i=1}^s f_i(Y, X_i)$$

$$\text{such that } g_{oj}(Y) \geq 0, \quad j = 1, \dots, n_o \quad [3.2]$$

$$g_{ij}(Y, X_i) \geq 0, \quad i = 1, \dots, s \quad j = 1, \dots, n_j$$

$$h_{ij}(Y, X_i) = 0, \quad i = 1, \dots, s \quad j = 1, \dots, n_e$$

The equality constraints h_{ij} usually define the relationship of the decoupling variables Y and the original X_i .

For example consider the problem (Figure 1) considered by Sobieski and coworkers (Ref 33 and 35). The detailed dimensions of each beam element whose cross section was an I-section are the detail design variables. With these detailed design variables the problem is fully coupled in that a change in a detail of one beam influences the stresses in the other two beams. However, it is possible to decouple the problem by using the cross sectional area, A , and the moment of inertia, I , of the beam. With these two variables defined for each beam, it is possible to determine the global response of the complete structure, i.e., the displacements and the forces that act between the beams. These two quantities can be considered as the decoupling variables,

GEOMETRY OF PORTAL FRAME EXAMPLE WITH I-BEAM CROSS SECTION

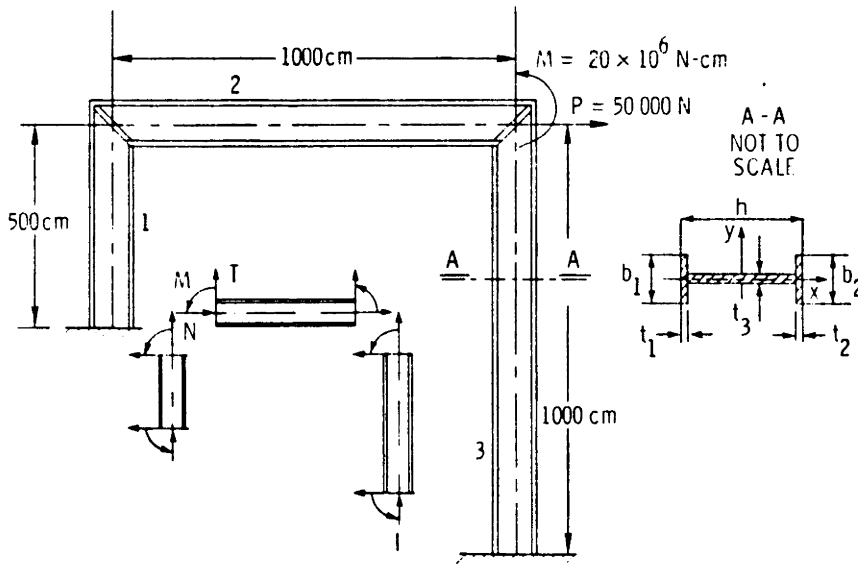


Figure 1. Geometry of Portal Frame with I-Beam Cross Section

because for a given vector of global or decoupling variables, the individual beams are fully decoupled, and can be independently analyzed for the effect of design details on stresses and local buckling. The equality constraints are used to relate A and I to the detail variables describing the beam cross section.

As shown later it is possible to take advantage of the decoupled form [3.2] by using a two-level optimization solution. However, even if a single-level solution is used, the decoupling process produces several computational advantages. One advantage of the decoupled system is that it is much cheaper to calculate derivatives of the constraints with respect to the design variables. The decoupling implies that changes in the local variables X_i of one subsystem do not cause any changes in all other subsystems as long as the decoupling variables Y are fixed. Therefore, derivatives with respect to the X_i variables usually become much cheaper to calculate. For example, in the frame problem if details of one beam cross section are changed without changing A and I only stresses in that beam are affected and they can be calculated without an overall analysis of the frame.

A second advantage of the decoupled system is that the single-level solution process can become cheaper by taking advantage of the decoupling. This is demonstrated here for the case of a penalty function optimization procedure coupled with Newton's method for unconstrained minimization (Ref 48).

3.2 Penalty Function Equivalent of Coupled System

Let us now consider the penalty function equivalent of the coupled [3.1] system. From equations [3.1] and [2.2] the total function φ is

$$\varphi(\mathbf{X}, r) = f(\mathbf{X}) + r \sum_{i=1}^m p(g_i) \quad [3.3]$$

For the above coupled system we have the following equation for the direction vector \mathbf{S}

$$\mathbf{H} \mathbf{S} = -\mathbf{G} \quad [3.4]$$

The above hessian matrix is fully coupled and, therefore, fully populated.

3.3 Penalty Function Equivalent of Decoupled System

For the decoupled system [3.2], the total function φ to be minimized is

$$\begin{aligned} \varphi(\mathbf{Y}, \mathbf{X}_i) = & f_o(\mathbf{Y}) + \sum_{i=1}^s f_i(\mathbf{Y}, \mathbf{X}_i) + r \sum_{j=1}^{n_o} p[g_{oj}(\mathbf{Y})] \\ & + r \sum_{i=1}^s \sum_{j=1}^{n_i} p[g_{ij}(\mathbf{Y}, \mathbf{X}_i)] \\ & + \frac{\beta}{\sqrt{r}} \sum_{i=1}^s \sum_{j=1}^{n_e} q[h_{ij}(\mathbf{Y}, \mathbf{X}_i)] \end{aligned} \quad [3.5]$$

We rewrite equation [3.5] as

$$\varphi(\mathbf{Y}, \mathbf{X}_i) = \varphi_o(\mathbf{Y}) + \sum_{i=1}^s \varphi_i(\mathbf{Y}, \mathbf{X}_i) \quad [3.6]$$

where

$$\varphi_o(\mathbf{Y}) = f_o(\mathbf{Y}) + r \sum_{j=1}^{n_o} p[g_{oj}(\mathbf{Y})] \quad [3.7]$$

and

$$\begin{aligned} \varphi_i(\mathbf{Y}, \mathbf{X}_i) = & f_i(\mathbf{Y}, \mathbf{X}_i) + r \sum_{j=1}^{n_i} p[g_{ij}(\mathbf{Y}, \mathbf{X}_i)] \\ & + \frac{\beta}{\sqrt{r}} \sum_{j=1}^{n_e} q[h_{ij}(\mathbf{Y}, \mathbf{X}_i)] \end{aligned} \quad [3.8]$$

Applying Newton's method directly to equation [3.5] for the minimization of φ , we have the following form of the equation for $\Delta\mathbf{Y}$ and $\Delta\mathbf{X}$

$$\begin{bmatrix} H_{oo} & H_{o1} & H_{o2} & \dots & H_{os} \\ H_{o1}^T & H_{11} & 0 & \dots & 0 \\ H_{o2}^T & 0 & H_{22} & \dots & 0 \\ & & & \vdots & \\ & & & & \\ H_{os}^T & 0 & 0 & \dots & H_{ss} \end{bmatrix} \begin{Bmatrix} \Delta\mathbf{Y} \\ \Delta\mathbf{X}_1 \\ \Delta\mathbf{X}_2 \\ \vdots \\ \Delta\mathbf{X}_s \end{Bmatrix} = - \begin{Bmatrix} \partial\varphi/\partial\mathbf{Y} \\ \partial\varphi/\partial\mathbf{X}_1 \\ \partial\varphi/\partial\mathbf{X}_2 \\ \vdots \\ \partial\varphi/\partial\mathbf{X}_s \end{Bmatrix} \quad [3.9]$$

where

$$\begin{aligned}
H_{oo} &= \frac{\partial^2 \phi}{\partial Y^2} = \frac{d^2 \phi_o}{dY^2} + \sum_{i=1}^s \frac{\partial^2 \phi_i}{\partial Y^2} \\
&= \frac{d^2 f_o}{dY^2} + r \sum_{j=1}^{n_o} \left[\frac{d^2 p}{dg^2} \left(\frac{\partial g_j}{\partial Y} \right)^T \left(\frac{\partial g_j}{\partial Y} \right) + \frac{dp}{dg} \frac{\partial^2 g_j}{\partial Y^2} \right] \\
&+ \sum_{i=1}^s \left[\frac{\partial^2 f_i}{\partial Y^2} + r \sum_{j=1}^{n_i} \left[\frac{d^2 p}{dg^2} \left(\frac{\partial g_{ij}}{\partial Y} \right)^T \left(\frac{\partial g_{ij}}{\partial Y} \right) + \frac{dp}{dg} \frac{\partial^2 g_{ij}}{\partial Y^2} \right] \right] \\
&+ \frac{\beta}{\sqrt{r}} \sum_{i=1}^s \sum_{j=1}^{n_e} \left[\frac{d^2 q}{dh^2} \left(\frac{\partial h_{ij}}{\partial Y} \right)^T \left(\frac{\partial h_{ij}}{\partial Y} \right) + \frac{dq}{dh} \frac{\partial^2 h_{ij}}{\partial Y^2} \right]
\end{aligned} \tag{3.10}$$

$$\begin{aligned}
H_{ii} &= \frac{\partial^2 \phi}{\partial X_i^2} = \sum_{i=1}^s \frac{\partial^2 \phi_i}{\partial X_i^2} \\
&= \sum_{i=1}^s \left[\frac{\partial^2 f_i}{\partial X_i^2} + r \sum_{j=1}^{n_i} \left[\frac{d^2 p}{dg^2} \left(\frac{\partial g_{ij}}{\partial X_i} \right)^T \left(\frac{\partial g_{ij}}{\partial X_i} \right) + \frac{dp}{dg} \frac{\partial^2 g_{ij}}{\partial X_i^2} \right] \right] \\
&+ \frac{\beta}{\sqrt{r}} \sum_{i=1}^s \sum_{j=1}^{n_e} \left[\frac{d^2 q}{dh^2} \left(\frac{\partial h_{ij}}{\partial X_i} \right)^T \left(\frac{\partial h_{ij}}{\partial X_i} \right) + \frac{dq}{dh} \frac{\partial^2 h_{ij}}{\partial X_i^2} \right]
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
H_{oi}^T &= \frac{\partial^2 \phi}{\partial X_i \partial Y} = \sum_{i=1}^s \frac{\partial^2 \phi_i}{\partial X_i \partial Y} \\
&= \sum_{i=1}^s \left[\frac{\partial^2 f_i}{\partial X_i \partial Y} + r \sum_{j=1}^{n_i} \left[\frac{d^2 p}{dg^2} \left(\frac{\partial g_{ij}}{\partial X_i} \right)^T \left(\frac{\partial g_{ij}}{\partial Y} \right) + \frac{dp}{dg} \frac{\partial^2 g_{ij}}{\partial X_i \partial Y} \right] \right] \\
&+ \frac{\beta}{\sqrt{r}} \sum_{i=1}^s \sum_{j=1}^{n_e} \left[\frac{d^2 q}{dh^2} \left(\frac{\partial h_{ij}}{\partial X_i} \right)^T \left(\frac{\partial h_{ij}}{\partial Y} \right) + \frac{dq}{dh} \frac{\partial^2 h_{ij}}{\partial X_i \partial Y} \right]
\end{aligned} \tag{3.12}$$

In the above expressions approximate second derivatives are obtained by neglecting the second derivatives of the constraints with respect to the design variables.

3.4 Benefits of Decoupling in the Solution Process

Equation [3.9] that needs to be solved for the direction vector in the Y and X_j space is block-sparse due to the lack of coupling between the subsystems, i.e., $\frac{\partial^2 \phi_k}{\partial X_i \partial X_j} = 0$ for $i \neq k$ and $j \neq k$. Savings in storage and computation of the hessian matrix can be achieved by taking advantage of the sparsity. The terms in the hessian matrix as given by equations [2.12] and [2.13] need not be summed over all the constraints, but only over those affected by changes in the design variable being perturbed.

The magnitude of the computational savings is a function of the total number of subsystems and the number of local retained variables in each subsystem. The larger these numbers are, the greater the benefits of using the decoupled system. For typical large engineering systems, where multi-level applications are being suggested, the values of these numbers are usually high, and such systems are, therefore, ideally suited for the proposed scheme of solution. There are three main sources of savings that can be identified: (i) savings in computing the terms of the hessian matrix, (ii) savings in computing the constraint gradients due to fewer global analyses since changes in local variables do not affect global quantities, and (iii) savings in the solution of equation [3.9] if the sparsity of this equation is taken into account. Let us consider each of these quantities.

3.4.1 Savings in Evaluation of Hessian Matrix

By noting that various terms of the hessian matrix in equation [3.9] are not affected by all design variables and constraints considerable savings can be achieved. For large systems the evaluation of the terms of this matrix can constitute a major portion of the optimization effort, and hence these savings can help reduce the computational effort by 30-50 percent.

If no allowance is made for the decoupled nature of the terms in the hessian the number of loops that must be performed to evaluate the hessian matrix is $NCON \times NDV \times NDV / 2$. On the other hand only global constraints affect the H_{oo} terms, while the H_{ji} and H_{oi}^T terms of a subsystem i are only affected by local constraints of that subsystem alone.

3.4.2 Savings in Constraint Gradients

For the decoupled system changes in local variables do not affect global quantities. Therefore when the derivatives of constraints with respect to the local variables are required there is no need to perform a global analysis. The details of computing these derivatives are discussed in Chapter 5.

If the total number of global and local design variables is n_g and n_l respectively, the computational savings resulting from not having to perform global analyses and using the chain rule of differentiation to compute local constraints derivatives is given by

$$\left[1 - \frac{n_g + 1}{n_l + 1} \right] \times 100 \quad [3.13]$$

This is because instead of n_l global analyses only n_g analyses are required. The total amount of savings is dependent on how expensive a global analysis of the system is compared to the cost of optimization.

For most realistic problems where the number of design variables is very large, this saving, though large, can be completely overshadowed by the savings resulting from not considering the contributions to the hessian matrix between the uncoupled individual subsystems.

3.4.3 Savings in Solution of Direction Equation

Another feature of the decoupled system is that the solution for ΔY and ΔX_i can be obtained with less computation than that for a coupled system. To do so, we note that from equation [3.9]

$$\mathbf{H}_{oo} \Delta Y = - \frac{\partial \phi}{\partial Y} - \sum_{i=1}^s \mathbf{H}_{oi} \Delta X_i \quad [3.14]$$

We also have

$$\mathbf{H}_{ii} \Delta X_i = - \frac{\partial \phi}{\partial X_i} - \mathbf{H}_{oi}^T \Delta Y \quad [3.15]$$

or

$$\Delta \mathbf{X}_i = - \mathbf{H}_{ii}^{-1} \left[\frac{\partial \varphi}{\partial \mathbf{X}_i} + \mathbf{H}_{oi}^T \Delta \mathbf{Y} \right] \quad [3.16]$$

Using equations [3.14] and [3.16] we have the following equation to be solved for $\Delta \mathbf{Y}$

$$\mathbf{H}^* \Delta \mathbf{Y} = - \frac{\partial \varphi}{\partial \mathbf{Y}} + \sum_{i=1}^s \mathbf{H}_{oi} \mathbf{H}_{ii}^{-1} \frac{\partial \varphi}{\partial \mathbf{X}_i} \quad [3.17]$$

where

$$\mathbf{H}^* = \mathbf{H}_{oo} - \sum_{i=1}^s \mathbf{H}_{oi} \mathbf{H}_{ii}^{-1} \mathbf{H}_{oi}^T \quad [3.18]$$

Thus, instead of solving equation [3.9] directly for $\Delta \mathbf{Y}$ and $\Delta \mathbf{X}_i$, for very large systems it is more efficient to first solve equations [3.17] for $\Delta \mathbf{Y}$ and then use equation [3.16] to solve for $\Delta \mathbf{X}_i$.

For the examples considered in this research the direct solution of equation [3.9] was cheap compared to the total cost of optimization. It was therefore not necessary to implement the decomposition in the solution process.

The overall procedure for decoupling is very simple, in that no major modification of structure of the system into a hierarchical one is required. The modeling of a hierarchical system can however be fairly involved from the programming point of view.

4.0 Solution Techniques for Single-Level and Multi-Level Formulations

In this chapter we examine the effect of equality constraints on the optimization if they are not explicitly eliminated and are retained to maintain consistency between the two levels.

As an example we shall consider the minimum weight design of the portal frame shown in Figure 2 subject to displacement and stress constraints. This structure was first formulated and solved by Sobieski et al (Ref 33). The frame is made of three beams, and the cross sections of the beams were I-sections (see Figure 1) as considered by Sobieski. Six detailed design variables governed the size of the beam cross sections. The problem is fully coupled in that any of the details of each beam influences the global displacements and the stresses in the other beams by changing the internal load distribution.

The problem is decoupled by using a vector Y containing the area and moment of inertia of each beam. With this vector fixed, the internal loads in

the beams are fixed so that the stresses in each beam depend only on the detail section parameters of that beam. For an I-section the relationship between the detail design variables and the decoupling variables was simple enough so that two of the detail design variables could be eliminated and easily replaced by the decoupling variables. Explicit relations were obtained for the one-to-one correspondence relating the detail design variables to the two global variables, the cross sectional area A and the moment of inertia I , and vice versa. A more complicated section was chosen instead of the original I-section selected by Sobieski and coworkers and no attempt was made to develop similar explicit expressions relating the detail design variables to the global design variables. The beam cross section was now considered to be made up of 8 identical hat-stiffeners (see Figure 2(b)). Each beam has six dimensional variables governing the cross sectional shape of the hats (Figure 2(c)).

However, additional equality constraints must be imposed on the problem to maintain consistency between the detail design variable vectors X_j and the global variables Y . For example, the constraint on the cross sectional area, A , of each beam is (see Figure 2(c))

$$h = A - (d_1d_3 + d_2d_4 + d_5d_6) = 0 \quad [4.1]$$

There are several alternate formulations for solving equation [3.2] for the minimum weight of the frame. One possibility, which was considered in the present work is to ignore the hierarchical structure and solve [3.2] as a

GEOMETRY OF PORTAL FRAME EXAMPLE WITH HAT-STIFFENER CROSS SECTION

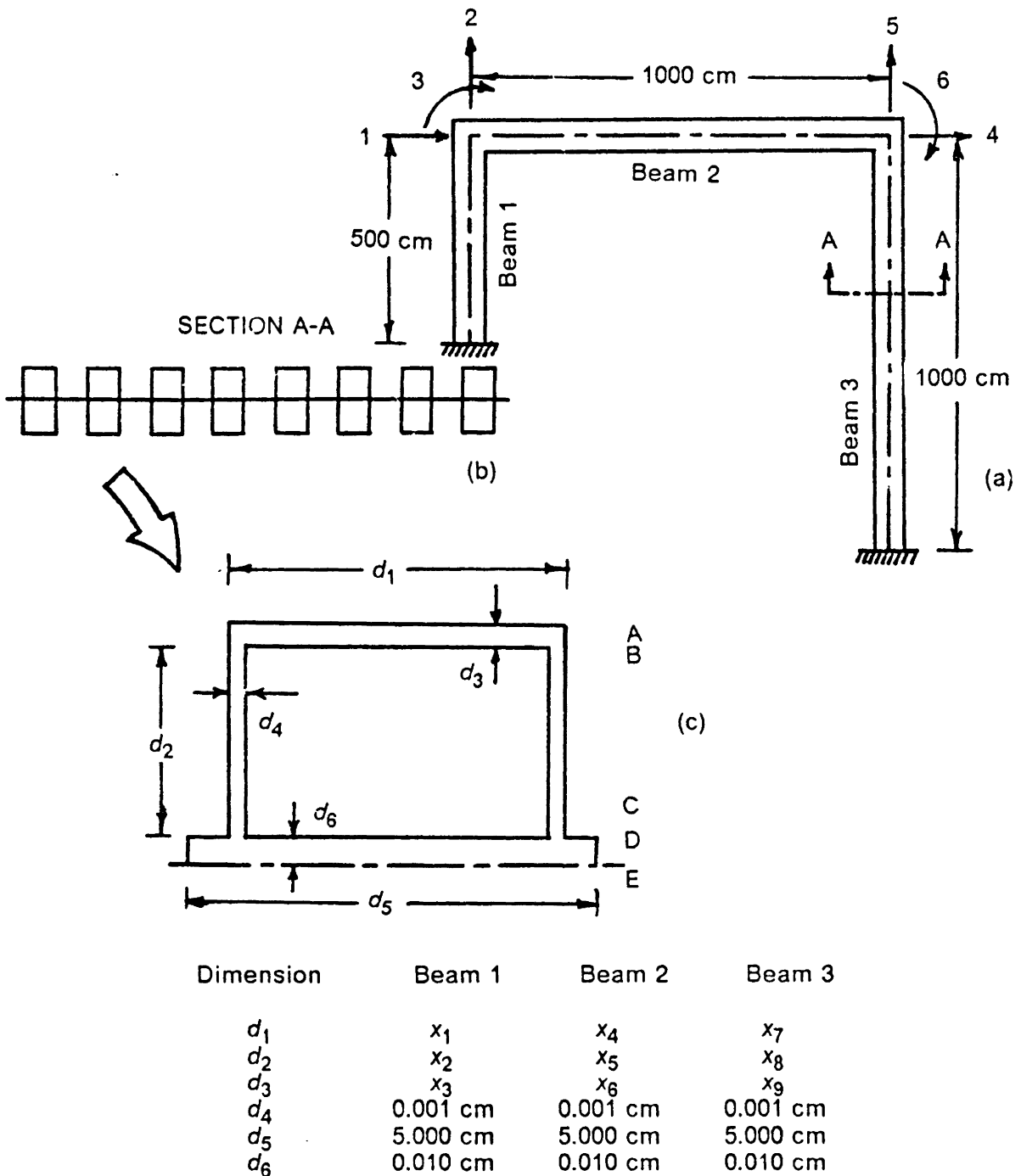


Figure 2. Geometry of Portal Frame with Hat-Stiffener Cross Section

single-level problem. This procedure is referred to as a two-level formulation with a single-level solution. The other approach used herein is to use a two-level solution technique.

There are several techniques for multi-level solution. The one selected here is a variant of the method suggested by Schmit and coworkers (Ref 28 and 30). For the portal frame example, the volume depends only on the areas so that $f_j(\mathbf{Y}, \mathbf{X}_j) = 0$, that is, the local design variables do not contribute directly to the objective function. The two-level formulation used was

$$\text{minimize } f = f_o(\mathbf{Y})$$

$$\text{such that } g_{oj}(\mathbf{Y}) \geq 0, \quad j = 1, \dots, n_o \quad [4.2]$$

$$g_{li}(\mathbf{Y}) \leq g_0, \quad i = 1, \dots, s \quad [4.2a]$$

where g_0 has a positive value given by equation [2.3]. During optimization the value of g_0 is gradually reduced so that it approaches zero at the end of optimization. The values of the s inequality constraints at the upper level $g_{li}(\mathbf{Y})$ are obtained from the solution of s lower level optimization problems

$$\text{minimize } g_{li}(\mathbf{Y}) = \sum_{j=1}^{n_e} h_{ij}^2(\mathbf{Y}, \mathbf{X}_j) \quad [4.3]$$

$$\text{such that } g_{ij}(\mathbf{Y}, \mathbf{X}_j) \geq 0, \quad i = 1, \dots, s \quad j = 1, \dots, n_j$$

That is, at the lower level the sum of the squares of the equality constraints is minimized for fixed values of the vector Y .

There are two possible sources of differences between the results from the single-level and two-level formulations. One is the change in problem formulation (adding the Y variables and the equality constraints). The other is the change in solution technique since the two-level optimization procedure is different from the single-level one. By using a single-level solution technique for the two-level formulation, we focus on the effect of the additional global variables and equality constraints. The single-level solution technique was repeated with several optimization algorithms available in NEWSUMT-A (Ref 48) and ADS (Ref 50) programs.

Because of programming complexity the multi-level solution technique was implemented only with NEWSUMT-A. Therefore, the effect of the two-level solution technique is not established as well as the effect of the change in formulation.

4.1 *Single-Level Formulation*

For the single-level formulation (equation [3.1]) all the detail design variables are considered as the variables to be optimized such that the weight of the frame is minimized. Similarly, all the constraints are considered at the same level in a constraint vector that consists of the global and the local

constraints of all the subsystems. This formulation is easy to set up, but the structural system is fully coupled and the hessian is fully populated. The cross sectional area A and the moment of inertia I are not considered by the optimizer at all. Thus no benefits of the decoupling variables are utilized in this solution scheme.

4.2 Single-Level Solution with Equality Constraints

For the single-level solution of equation [3.2] in addition to the detail design variables the decoupling variables A and I are also considered as design variables. Thus, the design vector consists of the decoupling or global design variables and the local design variables of all the subsystems. Similarly, all the constraints are considered at a single-level, so that the constraint vector consists of the global and the local constraints of all the subsystems. In addition to ensure consistency between the decoupling and the detail variables, 6 equality constraints are added to equate the cross sectional area A and the moment of inertia I of each beam to that obtained from the detail design variables.

4.3 Multi-Level Solution

The two-level solution follows the formulation of equations [4.2] and [4.3]. Each subsystem is considered in isolation at a lower level for fixed values of the global design variables. The local objective function is to minimize the difference in the quantities obtained from the global and local levels as defined by the squares of the equality constraints as in equation [4.1]. The constraints at the lower level are stresses and local buckling constraints.

At the higher level the objective is to minimize the overall weight of the structure as defined by the global design variables. The constraints at this level are global displacements and global buckling of the structure. In addition to these constraints additional constraints of the form of equation [4.2a] are added one for each subsystem to pass on the degree of satisfaction of the subsystem optimization, i.e., the lower level objective function (see equation [4.3]).

The optimization proceeds from the global to the local level, and the whole process is repeated until convergence is achieved.

4.4 Description of Frame Example

As an example consider the minimum volume design of the portal frame shown in Figure 2. Previous investigators (Refs 17, 26, 33 and 35) considered this portal frame with I-section beams as its three members. For an I beam with 6 cross sectional parameters and two global parameters A and I , it was possible to replace two local parameters by the global parameters. In order to consider a more complicated cross section, hat-stiffened beams are assumed instead of the I-section beam. Each beam is considered to be 40 cm wide with 8 identical symmetric hat-stiffeners as shown in Figure 2(b). Of the six dimensional parameters which govern the cross section of each beam, the last three are kept constant at the lower bound as shown in the figure. This leaves a total of 9 local design variables. Two different load conditions are considered to act on the portal frame, which is required to satisfy displacement and stress constraints for each load condition separately as shown in Table 1. The first 8 constraints are displacement constraints in degrees of freedom 4 and 6 (see Figure 2) due to load conditions 1 and 2. The rest of the 54 constraints are stress constraints, 18 for each beam. There are 4 normal stresses and 5 shear stresses monitored at various heights (A, B, C, D and E in Figure 2(c) of the hat-stiffener for each of the two load conditions.

Thus, for a coupled single-level optimization we minimize the volume of the portal frame with 9 design variables subject to 62 inequality constraints. Each of the design variables influences each of the displacement and stress

Table 1. Portal Frame Example - Design Information

Material	:	Aluminium
Young's modulus	:	7.06E6 N/cm ²
Normal stress limit	:	20000 N/cm ²
Shear stress limit	:	11600 N/cm ²
Number of load conditions	:	2

Load Case	Degree of Freedom	Load
1	4	5.0E4 N
2	6	-2.0E7 N-cm

Degree of Freedom	Maximum Allowable Displacement
4	± 4.0 cm
6	± 0.015 radians

Dimension Variable	Design Variable	Initial Value (cm)	Lower Bound (cm)	Upper Bound (cm)
d_1	x_1, x_4, x_7	1.0	0.1	5.0
d_2	x_2, x_5, x_8	50.0	10.0	-
d_3	x_3, x_6, x_9	0.25	0.01	-

constraints. As a result, to compute finite difference gradients of the constraints, a complete solution of the entire structure is required after perturbing each design variable in turn.

With decoupling or global design variables (the area A and the moment of inertia I of each beam), the formulation corresponding to problem [3.2] has 9 local and 6 global variables. The displacement constraints are global constraints as they depend only on the global design variables A and I and not on local details. On the other hand, the stress constraints in a beam are local constraints as they depend on the local details of the cross section of that beam. They also depend on the global design variables which determine the internal forces in each beam but not on the detail design variables of other beams. Equality constraints such as equation [4.1] ensure consistency of global and local design variables.

When we solve the resulting hierarchical problem as a single-level system with equality constraints we have a total of 15 design variables and 68 constraints (62 discussed previously and 6 equality constraints on the cross sectional area and moment of inertia of each beam). An advantage of this hierarchical approach is that the derivatives of the constraints with respect to the local design variables do not require a complete analysis of the full structure.

4.5 Optimization Techniques for the Frame Example

Two optimization packages were used to minimize the volume of the portal frame example discussed. The NEWSUMT optimization package (Ref 48) was modified to handle equality constraints (Ref 76 and 77). NEWSUMT is based on a quadratic extended interior penalty function (Ref 49) and Newton's method for unconstrained minimizations. The optimization is started with the penalty multiplier r at a value R_{init} which is changed after each unconstrained minimization by multiplication by a parameter R_{mult} .

The other optimization package used to minimize the volume of the portal frame was ADS Version 1.00 (Ref 50) developed by Vanderplaats. This program allows various optimizers to be used. The following strategies were attempted:-

1. Method of feasible directions for constrained minimization,
2. Modified method of feasible directions for constrained minimization,
3. Sequential unconstrained minimizations using the exterior penalty function method,
4. Sequential unconstrained minimizations using the linear extended interior penalty function method,
5. Sequential unconstrained minimizations using the quadratic extended interior penalty function method,
6. Sequential unconstrained minimizations using the cubic extended interior penalty function method, and

7. Augmented Lagrange multiplier method.

4.6 Results for the Frame Example

The results of the optimizations carried out by using NEWSUMT-A and various strategies with ADS are discussed.

A single-level formulation with a single-level solution using NEWSUMT-A resulted in an objective function of $2.1976 \times 10^4 \text{ cm}^3$ (Table 2). The value of the objective function was insensitive to the optimization parameters. For example, Table 2 shows very small variation of the final volume with R_{mult} which determines the reduction in penalty function multiplier at successive unconstrained minimizations. For the two-level formulation with a two-level solution, the objective function is about 25 percent higher (Table 2), and there is significant variation in the final volume when the value of R_{mult} is changed. For the two-level formulation with a single-level solution, the penalty function due to equality constraints is proportional to a multiplicative factor β (see equation [2.2]). NEWSUMT-A obtains the optimum volume with this formulation only for a narrow range of values of β (Table 2).

Results for the two-level formulation with a two-level solution were obtained from NEWSUMT-A only. This scheme requires a nested use of the optimizer, and ADS was considered to be too large and complex for such nested use.

Table 2. Optimization Results for Portal Frame Example, NEWSUMT-A Program

Single-level formulation with single-level solution

$RA_{\text{mult}} \bullet$	Objective Function, cm^3
0.1	2.19757E4
0.2	2.19754E4
0.3	2.19762E4

Two-level formulation with two-level solution

$RA_{\text{mult}} \bullet$	Objective Function, cm^3
0.1	2.75807E4
0.2	2.77585E4
0.3	2.83209E4

Two-level formulation with single-level solution

$\beta \blacksquare$	Objective Function, cm^3
1.0E1	diverging
1.0E2	diverging
1.0E3	2.19806E4
1.0E4	2.20263E4
1.0E5	2.28637E4
1.0E6	2.53047E4
1.0E7	2.70719E4
1.0E8	3.13687E4
1.0E9	3.73663E4

- multiplicative factor for penalty multiplier RA
 - see equation [2.2]
-
-

Using ADS with the method of feasible directions two distinct final designs with objective functions within about 2 percent were obtained for various values of THETAZ (the push-off factor) with the single-level formulation while the two-level formulation with a single-level solution gave much higher values of the objective function (Table 3). The single-level optimum could not be achieved for any value of THETAZ in the range that was attempted.

Using the modified method of feasible directions, even the single-level formulation did not converge to either of the solutions but was only 10-20 percent off. The two-level formulation with a single-level solution resulted in much higher values of the objective function (Table 4) for a wide range of THETAZ.

Various SUMT (Sequence of Unconstrained Minimization Techniques) variants were attempted with ADS (Tables 5, 6, 7, and 8). In most cases the single-level formulation converged to one of two previous optima for a wide range of optimization parameters, while the two-level formulation with a single-level solution was sensitive to optimization parameters and produced solutions that varied over a wide range.

Using ADS with the exterior penalty function method both the single-level and the two-level formulation with a single-level solution worked fairly well. Two distinct local minima were obtained depending on the choice of two optimization parameters (Table 5). These were R_{init} , the initial value of the penalty multiplier and R_{mult} , the factor by which r is multiplied after each unconstrained minimization.

Table 3. Optimization Results for Portal Frame Example, ADS Program, Method of Feasible Directions

Single-level formulation with single-level solution

THETAZ	Objective Function, cm ³	CPU Time *
0.1	2.25162E4	2.63
0.5	2.25366E4	1.03
1.0	2.19940E4	1.89
2.0	2.25219E4	1.88
10.0	2.25668E4	3.80

Two-level formulation with single-level solution

THETAZ	Objective Function, cm ³	CPU Time *
0.1	5.02288E4	0.35
1.0	4.65508E4	0.34
2.0	4.78523E4	0.40
0.5	4.92139E4	0.35
10.0	4.76949E4	0.75

* CPU seconds on IBM 3084

Table 4. Optimization Results for Portal Frame Example, ADS Program, Modified Method of Feasible Directions

Single-level formulation with single-level solution

THETAZ	Objective Function, cm ³	CPU Time *
0.1	2.70379E4	0.80
0.5	2.70926E4	0.96
1.0	2.72564E4	4.90
2.0	2.47454E4	4.81
10.0	2.71547E4	4.94

Two-level formulation with single-level solution

THETAZ	Objective Function, cm ³	CPU Time *
0.1	5.02288E4	0.35
0.5	4.92139E4	0.35
1.0	4.65508E4	0.34
2.0	5.00248E4	0.40
10.0	4.65403E4	0.39

* CPU seconds on IBM 3084

Table 5. Optimization Results for Portal Frame Example, ADS Program, SUMT - Exterior Penalty Function

Single-level formulation with single-level solution

RA_{init} ■	RA_{mult} ●	Objective Function, cm^3	CPU Time *
1.0	1.1	9.86136E4*	14.62
1.0	2.0	2.19403E4	21.45
1.0	5.0	2.19727E4	17.57
100.0	5.0	2.19727E4	16.65
40000.0	1.1	2.24680E4	29.96
40000.0	2.0	2.25118E4	29.40
40000.0	5.0	2.25136E4	16.89
30000.0	5.0	2.19736E4	23.44
50000.0	5.0	2.25131E4	15.53

Two-level formulation with single-level solution

RA_{init} ■	RA_{mult} ●	Objective Function, cm^3	CPU Time *
1.0	1.1	2.09600E3*	4.38
1.0	2.0	2.09600E3*	6.58
1.0	5.0	2.43817E3*	> 60
100.0	5.0	2.34965E4	18.75
40000.0	1.1	2.19718E4	36.95
40000.0	2.0	2.25028E4	23.59
40000.0	5.0	2.25118E4	22.95
40000.0	10.0	2.26015E4	27.24
30000.0	5.0	2.25370E4	24.43
50000.0	5.0	2.25123E4	21.15

- initial value of penalty multiplier RA
- multiplicative factor for penalty multiplier RA
- * constraints violated

* CPU seconds on IBM 3084

With linear (Table 6), quadratic (Table 7) and cubic (Table 8) extended interior penalty function methods, and the augmented Lagrange method (Table 9), the two-level formulation with a single-level solution resulted in substantially higher values of the objective function. In all cases the single-level results did not exhibit any convergence difficulties, while the two-level formulation with a single-level solution results were very sensitive to optimization parameters.

Figure 3 gives a graphical picture of the final optimum values of the volume of the frame which we seek to minimize. These are plotted as a function of the cross sectional area A of beam 2. The wide scatter of points indicates the range of values that can be obtained by using several optimization techniques. No trend is being established here, just the fact that the results are very sensitive to optimization parameters and result in a large range of optimum values.

These results indicate that using equality constraints to create a two-level formulation for the frame problem made the optimization solution very sensitive to the optimization parameters of NEWSUMT-A and ADS. Of all the schemes used, the two-level formulation with a two-level solution strategy required the longest computational times and did not obtain the single-level optimum for any value of the optimization parameters. The computational times are included in the tables to serve as a guide to compare the different schemes of solution, and also because in several cases a longer computational time indicates convergence difficulties.

Table 6. Optimization Results for Portal Frame Example, ADS Program, SUMT - Linear Extended Interior Penalty Function

Single-level formulation with single-level solution

RA_{init} ■	RA_{mult} ●	Objective Function, cm^3	CPU Time *
10.0	0.1	2.18909E4	13.17
10.0	0.2	2.18902E4	18.13
100.0	0.1	2.19735E4	28.50
100.0	0.2	2.19729E4	26.09

Two-level formulation with single-level solution

RA_{init} ■	RA_{mult} ●	Objective Function, cm^3	CPU Time *
10.0	0.1	3.35890E5	24.79
10.0	0.2	3.03494E4*	> 60
100.0	0.1	2.43565E3*	> 60
100.0	0.2	4.07846E4	57.30

- initial value of penalty multiplier RA
- multiplicative factor for penalty multiplier RA
- * constraints violated

* CPU seconds on IBM 3084

Table 7. Optimization Results for Portal Frame Example, ADS Program, SUMT - Quadratic Extended Interior Penalty Function

Single-level formulation with single-level solution

RA_{init} ■	RA_{mult} ●	Objective Function, cm^3	CPU Time *
3.333	0.1	2.25113E4	27.55
3.333	0.2	2.25107E4	24.76
3.333	0.3	2.25104E4	26.85
3.333	0.5	2.25089E4	25.36
3.333	0.8	2.25885E4	17.57
3.333	0.9	2.29631E4	6.77
100.000	0.2	2.25125E4	24.91

Two-level formulation with single-level solution

RA_{init} ■	RA_{mult} ●	Objective Function, cm^3	CPU Time *
3.333	0.1	2.33671E4	35.19
3.333	0.2	2.47664E4	41.77
3.333	0.3	2.48134E4	29.52
3.333	0.5	2.51340E4	45.31
3.333	0.8	2.48031E4	39.21
3.333	0.9	2.46492E4	35.89
100.000	0.2	6.42094E4	> 60

- initial value of penalty multiplier RA
- multiplicative factor for penalty multiplier RA

* CPU seconds on IBM 3084

Table 8. Optimization Results for Portal Frame Example, ADS Program, SUMT - Linear Extended Interior Penalty Function

Single-level formulation with single-level solution

RA_{init} ■	RA_{mult} ●	Objective Function, cm ³	CPU Time *
10.0	0.1	2.25089E4	15.25
10.0	0.2	2.25078E4	17.00
10.0	0.3	2.25038E4	22.51
10.0	0.5	2.25016E4	22.70
10.0	0.8	2.26330E4	21.74
10.0	0.9	2.33484E4	6.93
100.0	0.2	2.19722E4	20.07

Two-level formulation with single-level solution

RA_{init} ■	RA_{mult} ●	Objective Function, cm ³	CPU Time *
10.0	0.1	1.66837E4*	50.97
10.0	0.2	2.77137E4*	> 60
10.0	0.3	3.58748E4	36.78
10.0	0.5	3.94636E4	44.33
10.0	0.8	3.41127E4	43.40
10.0	0.9	2.21024E4	47.03
100.0	0.2	2.37258E3*	> 60

- initial value of penalty multiplier RA
- multiplicative factor for penalty multiplier RA
- * constraints violated

* CPU seconds on IBM 3084

Table 9. Optimization Results for Portal Frame Example, ADS Program, Augmented Lagrange Multiplier

Single-level formulation with single-level solution			
RA_{init} ■	RA_{mult} ●	Objective Function, cm ³	CPU Time *
100.0	5.0	2.19686E4	9.57
3333.333	2.0	2.19672E4	10.48
3333.333	5.0	2.25200E4	7.16
3333.333	10.0	2.19742E4	20.51
Two-level formulation with single-level solution			
RA_{init} ■	RA_{mult} ●	Objective Function, cm ³	CPU Time *
100.0	5.0	2.36540E4	23.47
3333.333	2.0	2.36698E4	18.26
3333.333	5.0	2.19684E4	20.26
3333.333	10.0	2.25167E4	23.37

■ initial value of penalty multiplier RA
 ● multiplicative factor for penalty multiplier RA

* CPU seconds on IBM 3084

OPTIMUM VOLUME FOR PORTAL FRAME EXAMPLE

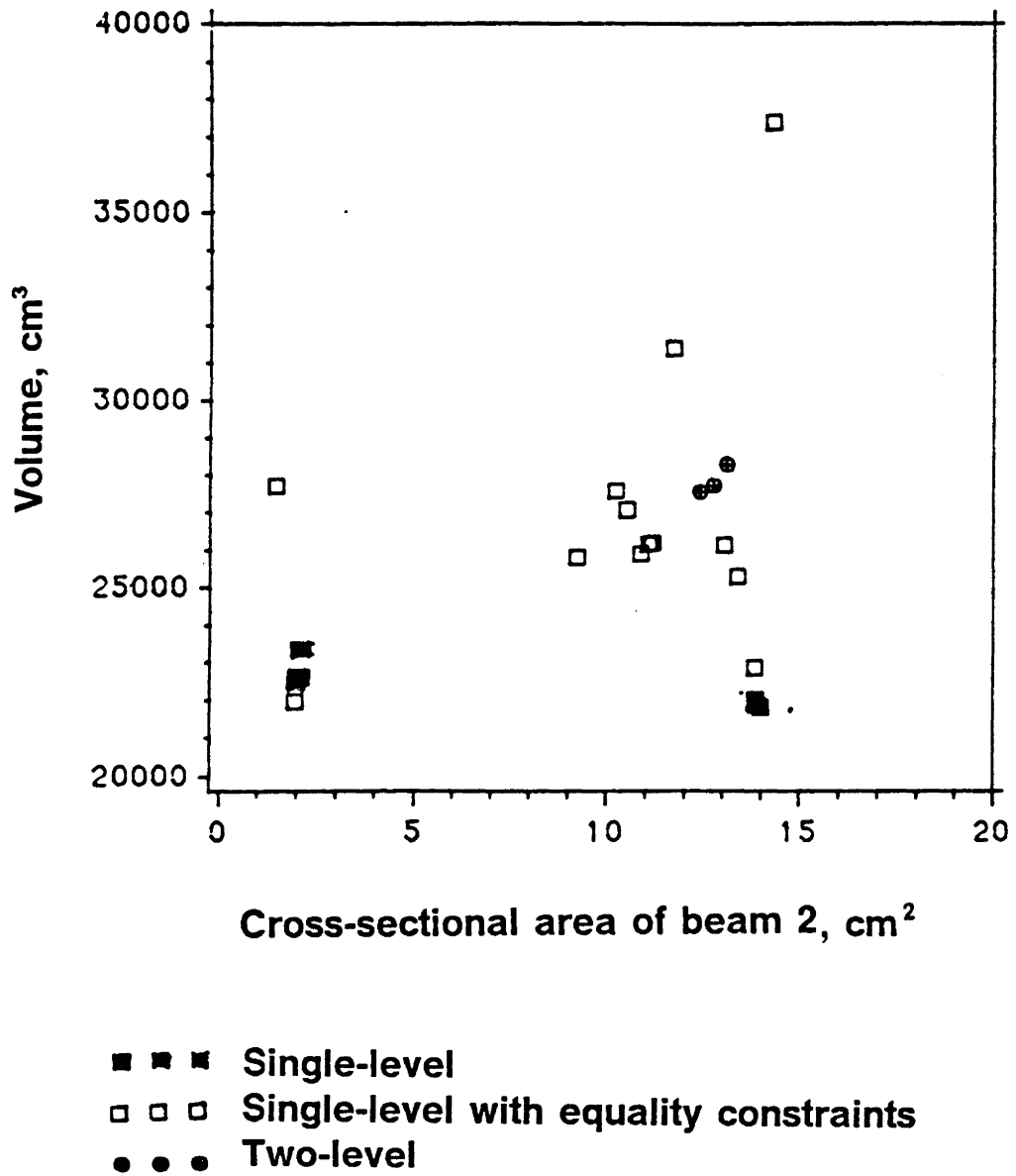


Figure 3. Optimum Volume as a Function of the Cross Sectional Area of Beam 2 for all Optimization Techniques

The optimality of several solutions obtained for the two-level formulations was checked by restarting the final solution using the single-level solution. In all cases one of the two single-level optima reported was obtained. This suggests that these two-level solutions were not local minima, but reflect numerical difficulties associated with the additional global variables and the presence of equality constraints.

4.7 Conclusions for the Frame Example

In multi-level optimization the optimized system is decomposed into a hierarchy of subsystems with greater and greater detail in the analysis focus. Equality constraints on various design quantities are often used in order to coordinate the values at successive levels and ensure consistency between levels. Typically these equality constraints have been imposed by variable elimination at the lower levels. In complex systems this is not always possible or easy to implement, and these equality constraints may have to be retained in the optimization process. In this chapter the impact of such a retention has been examined for a simple portal frame problem. The equality constraints introduce numerical difficulties, and the problem becomes very sensitive to optimization parameters for a wide range of optimization algorithms.

Beside the numerical difficulties, the use of global variables and equality constraints inflates the total number of design variables and constraints. It is therefore desirable to find an alternative to the use of equality constraints.

The next chapter describes a new technique that utilizes the hierarchical nature of the problem to reduce the computational effort but does not require equality constraints, or a multi-level solution. The key idea is to introduce global variables as intermediate variables, and use them to reduce the cost of gradient and search direction computations. The advantage is that the solution scheme benefits from the decoupled form of the hessian during both evaluation and solution. This scheme attempts to combine the advantages of both the single-level and the multi-level schemes. Though this technique cannot be used for problems that cannot be handled by single-level schemes, and must be solved by multi-level schemes, the method can successfully be applied to a large variety of large structural optimization problems. The discussion of the new technique is limited to using a penalty function approach, though the strategy can easily be modified for other optimization techniques.

5.0 Solution using Decoupling Intermediate Variables

In the previous chapter we considered the use of equality constraints for creating a hierarchical optimization problem and found that this can cause severe numerical difficulties. These numerical difficulties are associated apparently with the violation of the constraints during straight-line moves in the design space. Additionally, the use of equality constraints increases the number of design variables because both local and global variables are retained. It is possible to reduce the number of variables by using a local linearization of the equality constraints to eliminate some of the local variables. However, such a technique would still be expected to have numerical difficulties associated with constraint violation at the end of each one-dimensional move.

In this work an alternate approach is proposed where the global variables Y are used only as intermediate variables for the purpose of reducing the cost

of the optimization process. The original design variables are retained for the purpose of the optimization, which is carried out at a single-level. The advantage of this technique is that problems associated with multi-level optimization and problems attributed to the use of equality constraints are avoided. However, all the computational advantages of the multi-level formulation are still retained.

The proposed technique has two stages. In the first stage the equality constraints are linearized and used to eliminate some of the local variables. The system consisting of the global variables and the remaining local variables is decoupled in the sense that local variables affect only local constraints. The gradient and second-derivative calculations are inexpensive in this decoupled system. The second stage consists of a transformation of the gradient and second derivatives from the decoupled system to the original system. This transformation which is shown to be trivial permits us to carry out the optimization in the original design space while still enjoying the computational benefits of a hierarchical formulation.

The first stage of the proposed technique begins with the equality constraints, equation [3.2]

$$h_{ij}(\mathbf{Y}, \mathbf{X}_j) = 0, \quad i = 1, \dots, s \quad j = 1, \dots, n_e$$

We consider these constraints as implicit equations for n_e of the components of the \mathbf{X}_j in terms of the vector \mathbf{Y} and the remaining components of \mathbf{X}_j . For a general complex case, it is not possible to do this explicitly, but an explicit

solution is not required for our purpose. For the case considered here we make a simplifying assumption which is not essential in general; we assume that the global variables can be expressed in terms of the local variables as in equation [4.1]. For example, for the frame problem, the cross section area A is expressed in terms of the local variables by the constraint

$$h = A - (d_1d_3 + d_2d_4 + d_5d_6) = 0$$

Thus the equality constraints for the i -th subsystem define a subset D_i of order n_e of the vector Y which can be expressed in terms of the local variables X_i . Since we plan to use the equality constraints to eliminate some of the local variables we denote those as E_i (the eliminated set) and the remaining variables of X_i as R_i (the retained set). The set of variables D_i (the decoupling set) are of the same order as E_i (both are of order n_e). The equality constraint is therefore a relation of the form

$$D_i = D_i(R_i, E_i) \tag{5.1}$$

In general equation [5.1] cannot be explicitly solved for E_i , so that it is considered to be only an implicit relation for the E_i . However, it is assumed that the relation is invertible in the sense that the mapping from the D_i to the E_i variables is one-to-one, onto and continuous. We also assume that the D_i 's do not intersect, that is, each subsystem has its own global or decoupling variables, and there is possibly a vector D_0 which is the part of Y which is not associated with any subsystem. The objective function may be written as

$$f = f_o (D_o, D_1, D_2, \dots, D_s) + \sum_{j=1}^s f_j (D_j, R_j) \quad [5.2]$$

and the constraints as

$$\begin{aligned} g_{oj} (D_o, D_1, D_2, \dots, D_s) &\geq 0, & j = 1, \dots, n_o \\ g_{ij} (D_j, R_j) &\geq 0, & i = 1, \dots, s, \quad j = 1, \dots, n_j \end{aligned} \quad [5.3]$$

Because of the decoupled form of the objective function and constraints, their derivatives are inexpensive to calculate. In the next section we derive the transformations which may be used to calculate the derivatives of the original system from the derivatives of the decoupled system.

5.1 Derivative Transformation

Consider a general function

$$f = f (X_1, X_2, \dots, X_s) \quad [5.4]$$

We need to compute the derivatives of f with respect to all its variables.

The above function can be rewritten as

$$f = f_o (X_1, X_2, \dots, X_s) + \sum_{j=1}^s f_j (E_j, R_j) \quad [5.5]$$

Introducing the decoupling variables \mathbf{D} we can write the function in equation [5.5] as

$$f = f_0(\mathbf{X}_0, \mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_s) + \sum_{i=1}^s f_i(\mathbf{D}_i, \mathbf{R}_i) \quad [5.6]$$

For simplicity, let us start by considering the case of a single subsystem. Let f be the function in the coupled $\mathbf{E} - \mathbf{R}$ space, and let \bar{f} be the function in the decoupled $\mathbf{D} - \mathbf{R}$ space. Then we have

$$f_j(\mathbf{X}_j) = f_j(\mathbf{E}_j, \mathbf{R}_j) = \bar{f}_j(\mathbf{D}_j, \mathbf{R}_j)$$

For convenience we drop the index i , to get

$$f(\mathbf{X}) = f(\mathbf{E}, \mathbf{R}) = \bar{f}(\mathbf{D}, \mathbf{R})$$

and

$$\mathbf{D} = \mathbf{D}(\mathbf{R}, \mathbf{E})$$

We denote the number of variables of \mathbf{X} , \mathbf{D} , \mathbf{R} and \mathbf{E} as n_x , n_d , n_r and n_e , respectively. Obviously, $n_x = n_r + n_e$, and $n_d = n_e$. We will assume that f and \bar{f} are continuous functions and so are their first and second derivatives with respect to each element of the vectors \mathbf{R} , \mathbf{E} and \mathbf{D} .

Defining the design variable vector \mathbf{X} in the $\mathbf{E} - \mathbf{R}$ space as

$$\mathbf{X}^T = [\mathbf{E}, \mathbf{R}]^T$$

and the design variable vector \mathbf{X}^* in the transformed $\mathbf{D} - \mathbf{R}$ space as

$$\mathbf{X}^{*T} = [\mathbf{D}, \mathbf{R}]^T$$

Let us first define some notation for the equations that are subsequently developed. f is a function, and \mathbf{X}_1 and \mathbf{X}_2 are vectors. The derivative $f_{\mathbf{X}_1}$ is a column vector with the i -th component $\frac{\partial f}{\partial X_{1i}}$. Thus, we have

$$f_{\mathbf{R}} = \frac{\partial f}{\partial \mathbf{R}} \quad f_{\mathbf{E}} = \frac{\partial f}{\partial \mathbf{E}} \quad f_{\mathbf{D}} = \frac{\partial f}{\partial \mathbf{D}}$$

$$\bar{f}_{\mathbf{R}} = \frac{\partial \bar{f}}{\partial \mathbf{R}} \quad \bar{f}_{\mathbf{E}} = \frac{\partial \bar{f}}{\partial \mathbf{E}} \quad \bar{f}_{\mathbf{D}} = \frac{\partial \bar{f}}{\partial \mathbf{D}}$$

Similarly, the second derivatives of f form a matrix $f_{\mathbf{X}_1 \mathbf{X}_2}$ such that the element $\frac{\partial^2 f}{\partial X_{1i} \partial X_{2j}}$ is in the i -th row and the j -th column of the matrix. Thus, we have

$$f_{\mathbf{RR}} = \frac{\partial^2 f}{\partial \mathbf{R}^2} \quad f_{\mathbf{RE}} = \frac{\partial^2 f}{\partial \mathbf{R} \partial \mathbf{E}} \quad f_{\mathbf{EE}} = \frac{\partial^2 f}{\partial \mathbf{E}^2}$$

$$\bar{f}_{\mathbf{RR}} = \frac{\partial^2 \bar{f}}{\partial \mathbf{R}^2} \quad \bar{f}_{\mathbf{RD}} = \frac{\partial^2 \bar{f}}{\partial \mathbf{R} \partial \mathbf{D}} \quad \bar{f}_{\mathbf{DD}} = \frac{\partial^2 \bar{f}}{\partial \mathbf{D}^2}$$

\mathbf{D} is a vector of decoupling variables. The derivative $\mathbf{D}_{\mathbf{X}_1}$ is a matrix such that an element in its i -th row and the j -th column is $\frac{\partial D_j}{\partial X_{1i}}$. Thus, we have

$$\mathbf{D}_{\mathbf{R}} = \frac{\partial \mathbf{D}}{\partial \mathbf{R}} \quad \mathbf{D}_{\mathbf{E}} = \frac{\partial \mathbf{D}}{\partial \mathbf{E}}$$

The second derivative of \mathbf{D} denoted by $\mathbf{D}_{\mathbf{x}_1 \mathbf{x}_2}$ is a tensor of order three such that its k, i, j -th element is

$$\mathbf{D}_{\mathbf{x}_1 \mathbf{x}_2 \text{ } kij} = \frac{\partial^2 \mathbf{D}_k}{\partial \mathbf{x}_{1i} \partial \mathbf{x}_{2j}}$$

We then have the following notation

$$\mathbf{D}_{\mathbf{RR}} = \frac{\partial^2 \mathbf{D}}{\partial \mathbf{R}^2} \quad \mathbf{D}_{\mathbf{RE}} = \frac{\partial^2 \mathbf{D}}{\partial \mathbf{R} \partial \mathbf{E}} \quad \mathbf{D}_{\mathbf{EE}} = \frac{\partial^2 \mathbf{D}}{\partial \mathbf{E}^2}$$

The derivatives of \bar{f} are easily obtained and we seek transformation rules to obtain the derivatives of f . Using chain rule differentiation

$$f_{\mathbf{R}} = \bar{f}_{\mathbf{R}} + \bar{f}_{\mathbf{D}} \mathbf{D}_{\mathbf{R}} \quad [5.7]$$

and

$$f_{\mathbf{E}} = \bar{f}_{\mathbf{D}} \mathbf{D}_{\mathbf{E}} \quad [5.8]$$

Differentiating equation [5.8] with respect to \mathbf{E} , we have

$$f_{\mathbf{EE}} = \mathbf{D}_{\mathbf{E}}^T \bar{f}_{\mathbf{DD}} \mathbf{D}_{\mathbf{E}} + \mathbf{Z}^{\mathbf{EE}}$$

where

$$\mathbf{Z}_{ij}^{\mathbf{EE}} = \sum_k \bar{f}_{\mathbf{D}_k} \mathbf{D}_{\mathbf{EE}kij}$$

If the equality constraint is linearized and the second term is neglected then

$$\bar{f}_{DD} = D_E^{-1T} f_{EE} D_E^{-1} \quad [5.9]$$

Also, differentiating equation [5.7] with respect to E , we have

$$f_{RE} = \bar{f}_{RD} D_E + D_R^T \bar{f}_{DD} D_E + Z^{RD} D_E$$

where

$$Z_{ij}^{RD} = \sum_k \bar{f}_{D_k} D_{RD_{kij}}$$

Neglecting the second derivatives of the equality constraints and using equation [5.9] we have

$$\bar{f}_{RD} = f_{RE} D_E^{-1} - D_R^T D_E^{-1T} f_{EE} D_E^{-1} \quad [5.10]$$

Also, differentiating equation [5.7] with respect to R , we have

$$f_{RR} = \bar{f}_{RR} + D_R^T \bar{f}_{DR} + Z^{RR} + \bar{f}_{RD} D_R + D_R^T \bar{f}_{DD} D_R$$

where

$$Z_{ij}^{RR} = \sum_k \bar{f}_{D_k} D_{RR_{kij}}$$

Neglecting the second derivatives of the equality constraints and substituting from equations [5.9] and [5.10] we have

$$\begin{aligned} \bar{f}_{RR} = & f_{RR} - f_{RE} D_E^{-1} D_R - D_R^T D_E^{-1T} f_{RE}^T \\ & + D_R^T D_E^{-1T} f_{EE} D_E^{-1} D_R \end{aligned} \quad [5.11]$$

These relations relate the first and second derivatives of f and \bar{f} .

5.1.1 Single Subsystem Transformation Matrix

For a single subsystem the decoupling or global variable D is a function of the local variables R and E , i.e.,

$$D = D(R, E)$$

Using the chain rule of differentiation we have the set of equations

$$\Delta R = \Delta R$$

$$\Delta D = \frac{\partial D}{\partial R} \Delta R + \frac{\partial D}{\partial E} \Delta E$$

or

$$\begin{Bmatrix} \Delta R \\ \Delta D \end{Bmatrix} = \begin{bmatrix} I & 0 \\ D_R & D_E \end{bmatrix} \begin{Bmatrix} \Delta R \\ \Delta E \end{Bmatrix}$$

and inverting the above matrix, we have

$$\begin{Bmatrix} \Delta R \\ \Delta E \end{Bmatrix} = [t] \begin{Bmatrix} \Delta R \\ \Delta D \end{Bmatrix} \quad [5.12]$$

where the transformation matrix t is

$$[t] = \begin{bmatrix} I & 0 \\ -D_E^{-1} D_R & D_E^{-1} \end{bmatrix} \quad [5.13]$$

This same transformation matrix t is used to relate the design variables in the coupled and uncoupled design spaces, that is

$$X = t X' \quad [5.14]$$

Note that the following relations hold

$$\begin{Bmatrix} f_R \\ f_E \end{Bmatrix} = [t]^T \begin{Bmatrix} \bar{f}_R \\ \bar{f}_D \end{Bmatrix} \quad [5.15]$$

and

$$\begin{Bmatrix} \bar{f}_R \\ \bar{f}_D \end{Bmatrix} = [t]^T \begin{Bmatrix} f_R \\ f_E \end{Bmatrix} \quad [5.16]$$

It is also easily shown from equation [5.13] that

$$[t]^T \begin{bmatrix} f_{RR} & f_{RE} \\ f_{RE}^T & f_{EE} \end{bmatrix} [t] = \begin{bmatrix} \bar{f}_{RR} & \bar{f}_{RD} \\ \bar{f}_{RE}^T & \bar{f}_{DD} \end{bmatrix} \quad [5.17]$$

if the relations based on equality constraint linearization are used. Equation [5.17] is used for the search direction calculations in Section 5.3.

5.1.2 Multi-Subsystem Equations

In the previous subsection we looked at the transformation matrix for a single subsystem. Let us now consider the entire structure, which consists of a vector of design variables X_0 that are not a part of any subsystem, and X_1, X_2, \dots, X_s which are the design variable vectors for each of the subsystems. The transformation matrix for the entire structure can be considered as an assembly of the above vectors. Let X be the design variable vector of the entire structure. Then we can write

$$X^T = [X_0, X_1, X_2, \dots, X_s]^T \quad [5.18]$$

Similarly for the gradient vector we have

$$G^T = [G_0, G_1, G_2, \dots, G_s]^T \quad [5.19]$$

The transformation matrix for transforming equation [5.5] to [5.6] can be written as a partitioned matrix with the following structure

$$T = \begin{bmatrix} I & & & & \\ & t_1 & & & \\ & & t_2 & & \\ & & & \ddots & \\ & & & & \ddots & \\ & & & & & t_s \end{bmatrix} \quad [5.20]$$

where t_1, t_2, \dots, t_s are transformation matrices for the subsystems each having the form of equation [5.13]. At the top of this matrix we have an identity matrix for those variables that do not belong to any of the subsystems, identified here as X_0 .

This transformation matrix T can be used for the transformation which is discussed in the next section. It must be pointed out that even though this transformation matrix is given here, it is not necessary to use this matrix, since it has very large dimensions. While actually using this transformation matrix, each subsystem can be transformed in isolation, thereby saving on computer memory. The entire transformation matrix is never really used, since the matrices are obtained directly and not through an actual transformation.

5.2 *Choosing the Eliminated Set*

Unless there is some reason to choose a particular subset of the local design variables to be eliminated, the following scheme can be used to do so. Let a subsystem have n_x local design variables. Also, let n_d be the number of decoupling variables corresponding to this subsystem. Clearly the number of local design variables that need to be eliminated is n_e , where $n_e = n_d$. Now consider the subsystem transformation matrix \mathbf{t} given by [5.13]. This matrix has n_x rows and n_x columns. Consider a reduced matrix consisting of the matrix obtained from \mathbf{t} with n_e rows and columns deleted. There are a large number of such possible reduced matrices depending on which rows and columns are deleted. We can choose that set of design variables as the eliminated set which has the smallest condition number (ratio of the largest to smallest eigenvalue of the corresponding reduced matrix). This technique ensures that we choose a subsystem transformation matrix that is well conditioned, thereby eliminating numerical difficulties in the transformation process.

5.3 Calculating a Search Direction

If the optimization were conducted in the decoupled design space \mathbf{X}^* then the hessian matrix \mathbf{H}^* and the gradient vector \mathbf{G}^* are calculated and a search direction \mathbf{S}^* is obtained from equation [2.5] as

$$\mathbf{H}^* \mathbf{S}^* = -\mathbf{G}^* \quad [5.21]$$

The corresponding equation for the original system (see equation [2.5]) is

$$\mathbf{H} \mathbf{S} = -\mathbf{G} \quad [5.22]$$

\mathbf{H}^* and \mathbf{G}^* are inexpensive to calculate compared to \mathbf{H} and \mathbf{G} , and also equation [5.21] is easy to solve because it can be decomposed into a large number of small subsystems while equation [5.22] is expensive to solve.

This section shows how we can transform equation [5.22] into a system that is identical or very similar to equation [5.21]. We start by noting that since the direction vectors \mathbf{S}^* and \mathbf{S} represent increments in \mathbf{X}^* and \mathbf{X} based on equations [5.21] and [5.22], respectively. Therefore, from equation [5.14]

$$\mathbf{S} = \mathbf{T} \mathbf{S}^* \quad [5.23]$$

Also, from equation [5.16]

$$\mathbf{G} = \mathbf{T}^T \mathbf{G}^* \quad [5.24]$$

Substituting for \mathbf{S} and \mathbf{G} from equations [5.23] and [5.24] in equation [5.22] and premultiplying by \mathbf{T}^T we obtain

$$\hat{\mathbf{H}} \mathbf{s}^* = -\mathbf{g}^* \quad [5.25]$$

where

$$\hat{\mathbf{H}} = \mathbf{T}^T \mathbf{H} \mathbf{T} \quad [5.26]$$

It is shown in Appendix A that $\hat{\mathbf{H}}$ has the same sparsity pattern as \mathbf{H}^* , that is, it is also partitioned so that it is easily decoupled and decomposed. Furthermore, the elements of $\hat{\mathbf{H}}$ can be obtained from the elements of \mathbf{H}^* by adding correction terms involving the second derivatives of the equality constraints. In the present work these second order terms are neglected so that $\hat{\mathbf{H}} = \mathbf{H}^*$. Thus, the transformation of equation [5.22] into equation [5.25] achieved the desired goal and resulted in a system of equations which is inexpensive to assemble and solve. It is not necessary to calculate $\hat{\mathbf{H}}$ from equation [5.26] and \mathbf{g}^* from equation [5.24] since these can be obtained directly from finite difference calculations in the decoupled design space, i.e., from \mathbf{H}^* and \mathbf{g}^* (the hessian and gradient with respect to the design variable vector \mathbf{X}^*). Then equation [5.21] is solved for the search direction \mathbf{s}^* in the decoupled design space. Once \mathbf{s}^* is computed, the search direction \mathbf{s} in the original design space is retrieved using equation [5.23].

One factor that causes differences in the search direction while using this transformation is that though the hessian matrix \mathbf{H} in equation [5.22] is exact,

in practice it is approximated from the first derivatives of the constraints with respect to the design variables using techniques discussed in Chapter 2. Also, errors caused by inaccuracies in the derivatives usually computed by finite-differences result in differences in the terms of the hessian matrix. In reality then instead of equation [5.22] the equation that is solved is

$$[\mathbf{H} + \mathbf{H}_c] \mathbf{s} = - \mathbf{G} \quad [5.27]$$

where \mathbf{H}_c is a diagonal matrix with terms $\epsilon \mathbf{H}_{jj}$ along the diagonal. These correction terms are used to make the original hessian matrix \mathbf{H} more diagonally dominant to prevent a singular matrix. Typical values of ϵ are in the range 0.01 - 0.05.

Thus equation [5.21] is replaced by

$$[\mathbf{H}^* + \mathbf{H}_c^*] \mathbf{s}^* = - \mathbf{G}^* \quad [5.28]$$

where $\mathbf{H}_c^* = \mathbf{T}^T \mathbf{H}_c \mathbf{T}$.

Here \mathbf{H}^* is obtained directly from finite difference calculations in the decoupled design space. It is then corrected by adding \mathbf{H}_c^* to it. The computation of \mathbf{H}_c^* is not very expensive since \mathbf{H}_c is a diagonal matrix and the transformation matrix \mathbf{T} is also block-diagonal.

This correction matrix \mathbf{H}_c^* is not always necessary but may be needed if either (i) the direction vector in the original coupled design space must be followed, or (ii) if the direction computed without the correction is not usable

in that the total function does not decrease in that direction. This property can be easily verified by ensuring that

$$\mathbf{G} \cdot \mathbf{S} < 0 \quad [5.29]$$

5.4 Calculation of Derivatives of Constraints

Let us now consider how to compute the derivatives of the constraints with respect to each element of the decoupled design vector \mathbf{X}^* by finite differences, since these are required in the evaluation of \mathbf{H}^* and \mathbf{G}^* (the hessian and gradient with respect to the design variable vector \mathbf{X}^*).

For the original single-level system with the coupled design vector \mathbf{X} a constraint g is computed for a nominal design and its value is denoted as g_0 . Then each design variable is perturbed by a small amount Δx in turn with all other elements of \mathbf{X} held constant to obtain a new value for the constraint g_+ . The derivative of the constraint with respect to x is then obtained as

$$\frac{dg}{dx} = \frac{g_+ - g_0}{\Delta x}$$

For the modified system corresponding to the decoupled design vector \mathbf{X}^* the procedure of evaluating the derivatives of the constraints is not as straight-forward because changes in the values of the local design variables \mathbf{E} and \mathbf{R} affect the values of the global design variables \mathbf{D} . Also, the value of

the objective function and the constraints in equation [3.1] are defined in terms of the design vector \mathbf{X} and not \mathbf{X}^* . For the evaluation of \mathbf{H}^* and \mathbf{G}^* we need to perturb the components of \mathbf{R} and \mathbf{D} to obtain the constraint derivatives.

5.4.1 Perturbing Elements of \mathbf{R}

Repeating equation [5.12] we have

$$\begin{Bmatrix} \Delta \mathbf{R} \\ \Delta \mathbf{E} \end{Bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{D}_E^{-1} \mathbf{D}_R & \mathbf{D}_E^{-1} \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{R} \\ \Delta \mathbf{D} \end{Bmatrix}$$

For each subsystem, while variables corresponding to the vector \mathbf{R} are perturbed one at a time, the perturbations in the vector \mathbf{E} of that subsystem are estimated as

$$\Delta \mathbf{E} = -\mathbf{D}_E^{-1} \mathbf{D}_R \Delta \mathbf{R} \quad [5.30]$$

Equation [5.30] ensures that while $\Delta \mathbf{R}$ is being perturbed $\Delta \mathbf{D} = \mathbf{0}$.

These perturbations are added to the original design vector \mathbf{X} , and the constraints are evaluated for this design. Finite difference derivatives are then evaluated from changes in the value of these constraints from their base values.

While the elements of the vector \mathbf{R} for each subsystem are being perturbed, the vector \mathbf{D} for that and all other subsystems is held constant.

Thus the global constraints and the local constraints of all other subsystems are not changed when an element of the local variable vector \mathbf{R} is perturbed. Therefore a complete analysis of the entire structure is not necessary. It is sufficient to analyze each substructure being perturbed in isolation while keeping the forces acting on it at their values obtained from the unperturbed base state.

This procedure of perturbing each element of the vector \mathbf{R} for any given system is then repeated for all subsystems. Each time a change in an element of one subsystem has no effect on all other subsystems, due to the decoupling present between subsystems.

5.4.2 Perturbing Elements of \mathbf{D}

Now consider perturbations in the vector \mathbf{D} for a subsystem. Then the perturbations in the vector \mathbf{E} of that subsystem are estimated from equation [5.12] as

$$\Delta \mathbf{E} = \mathbf{D}_E^{-1} \Delta \mathbf{D} \quad [5.31]$$

Equation [5.31] ensures that during these perturbations $\Delta \mathbf{R} = \mathbf{0}$.

Perturbing elements of the vector \mathbf{D} in one subsystem changes the constraint values of the entire system. Thus, a $\Delta \mathbf{D}$ changes the global constraints and also, as a result of load redistribution, the local constraints of all subsystems, including the same subsystem.

Once the derivatives of the constraints are evaluated for all subsystems in the decoupled design space \mathbf{X}^* , the hessian \mathbf{H}^* and \mathbf{G}^* are obtained by methods outlined in Chapter 2.

5.4.3 Computing Constraint Derivatives with Respect to Original Variables

In order to utilize the constraint approximations discussed in Chapter 2, it is also necessary to obtain the derivatives of the constraints corresponding to the elements of the untransformed vector \mathbf{X} for all subsystems as these variables are present in the original system and their derivatives have not been evaluated.

One option is to use the transformation in equation [5.15] for each constraint to change the derivatives from the $\mathbf{D} - \mathbf{R}$ space to the $\mathbf{E} - \mathbf{R}$ space. It is cheaper to use the chain rule of differentiation directly to obtain these derivatives from those obtained by finite difference perturbations of the global variables \mathbf{D} without having to perform additional analyses of the complete structure as follows

$$\frac{dg_j}{d\mathbf{X}_j} = \frac{\partial g_j}{\partial \mathbf{X}_j} + \sum_{k=1}^{m_j} \frac{\partial g_j}{\partial \mathbf{D}_k} \frac{d\mathbf{D}_k}{d\mathbf{X}_j} \quad [5.32]$$

This expression can be used to compute the derivatives of each constraint g_j with respect to each of the elements of the vector \mathbf{X} , x_j . The term

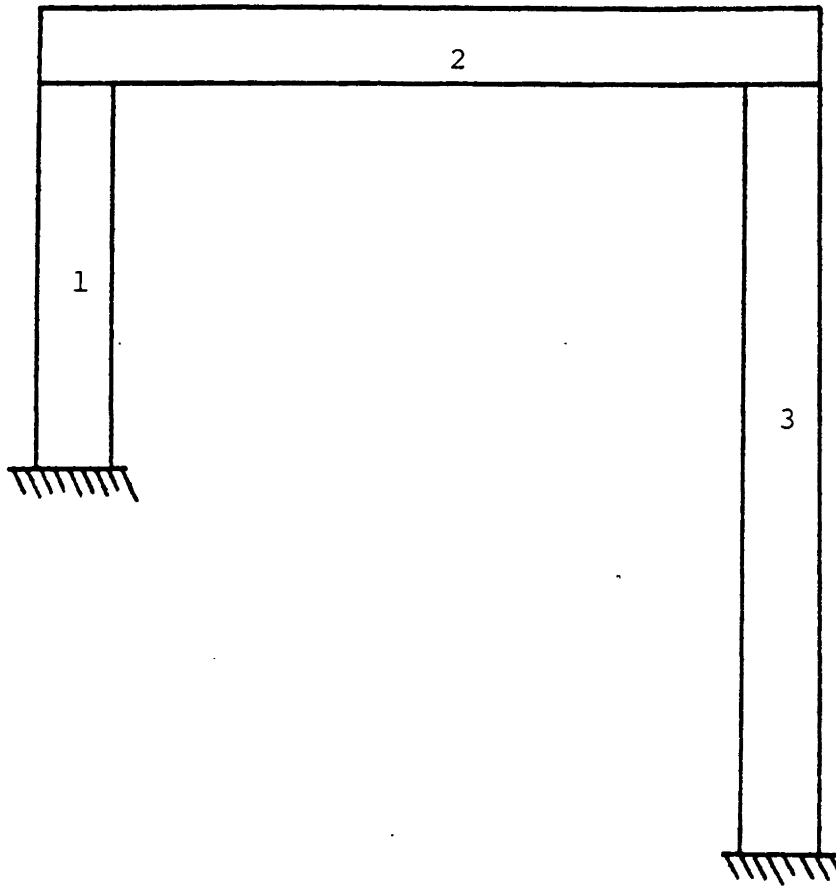
$\frac{\partial g_j}{\partial X_i} = \frac{\partial g_j}{\partial R_i}$ for elements of the retained variable vector, while it is zero for elements of the eliminated variable vector. $\frac{\partial g_j}{\partial R_i}$ and $\frac{\partial g_j}{\partial D_k}$ are already known from finite differences in the $\mathbf{D} - \mathbf{R}$ space, and $\frac{dD_k}{dX_i}$ is very easy to compute. As a result using equation [5.32] we can cheaply compute the derivatives in the original untransformed X_i space.

6.0 Numerical Results for Frame Example

In this chapter numerical results obtained by applying the proposed decoupling technique to the portal frame are presented. A variable cross section version of the portal frame example is considered, with each of the three beams modeled by several finite elements. The size of the problem is varied by changing the number of elements in each beam. It is shown that for small problems the costs associated with implementing the decoupling technique are not offset by the savings resulting from cheaper evaluation of the constraint derivatives and the hessian. But as the size of the problem is increased significant savings in computational costs are achieved.

The geometry of the portal frame with hat-stiffened cross section is shown in Figure 2. The loading consists of two loading conditions (Table 1). In Chapter 4 the last three dimensional variables $d_4 - d_6$ for each beam were kept constant at their lower bounds as shown in Figure 2. In this chapter only the fifth design variable d_5 is fixed at 5 cm (Figure 4) to preserve the overall

FINITE ELEMENT MESH FOR PORTAL FRAME (FRAME015)



Dimension	Beam 1	Beam 2	Beam 3
d_1	X_1	X_6	X_{11}
d_2	X_2	X_7	X_{12}
d_3	X_3	X_8	X_{13}
d_4	X_4	X_9	X_{14}
d_5	5.000 cm	5.000 cm	5.000 cm
d_6	X_5	X_{10}	X_{15}

Figure 3. Finite element model for portal frame FRAME015

width of the beams. The loading, global constraints on displacement and the design information for this case are summarized in Table 10. As discussed in Chapter 4, the portal frame is required to satisfy displacement and stress constraints for each load condition separately.

There are 8 displacement constraints on degrees of freedom 4 and 6 (see Figure 2) due to load conditions 1 and 2. There are 24 stress constraints at each beam cross section; 4 normal stresses and 5 shear stresses monitored at various heights (A, B, C, D and E in Figure 2(c)) of the hat-stiffener for each of the two load conditions. There are also 6 side constraints, 5 for the lower bounds on each of the design variables and 1 for the upper bound on d_1 . Three models of the frame are studied (see Figures 4, 9 and 14) designated as FRAME015, FRAME050 and FRAME125 with 15, 50 and 125 design variables respectively.

Decoupling was achieved by eliminating the design variables associated with d_1 and d_3 of Figure 2(c) and replacing them by the global variables, the cross sectional area A and the moment of inertia I .

The tables presented in this chapter focus on computational efficiency of the portal frame example. The total CPU time (presented here for an IBM 3084) consists of two main components: (i) the CPU time required to compute search directions displayed in the tables as Direction Time, and (ii) the CPU time for performing one-dimensional minimizations, displayed as ODM Time. The Direction Time further consists of three main components: (i) the CPU time required to compute the derivatives of the constraints displayed as Constraint

Table 10. Portal Frame Example - Design Information

Material	:	Aluminium
Young's modulus	:	7.06E6 N/cm ²
Normal stress limit	:	20000 N/cm ²
Shear stress limit	:	11600 N/cm ²
Number of load conditions	:	2

Load Case	Degree of Freedom	Load
1	4	5.0E4 N
2	6	-2.0E7 N-cm

Degree of Freedom	Maximum Allowable Displacement
4	± 4.0 cm
6	± 0.015 radians

Dimension Variable	Design Variable	Initial Value (cm)	Lower Bound (cm)	Upper Bound (cm)
d_1	x_1	1.0	0.1	5.0
d_2	x_2	50.0	10.0	-
d_3	x_3	0.25	0.01	-
d_4	x_4	0.025	0.001	-
d_6	x_5	0.5	0.01	-

Gradients Time, (ii) the CPU time required to evaluate the hessian matrix displayed as Hessian Setup Time, and (iii) the CPU time required to solve the direction equation displayed as Solution Time.

6.1 Numerical Results for Portal Frame FRAME015

The portal frame FRAME015 is designed for minimum volume at a single-level using the NEWSUMTA optimization program (Ref 76 and 77). The portal frame consists of three beams each of which is modeled as a single frame element (Figure 4). Thus, this model has three substructures that can be decoupled using global variables. Each element had 5 design variables controlling its cross sectional shape, for a total of 15 design variables. The number of global variables was 6. The number of constraints was 80, 8 global displacement constraints and 24 constraints on stresses and upper and lower bounds in each element.

Figure 5 shows the variation of the volume of the frame as a function of the CPU time used during optimization for exact constraints, and the linear, quadratic, reciprocal and conservative approximations of the constraints using the original coupled system. For this structure the reciprocal and the conservative approximations take substantially longer to converge to the weight obtained by the other methods. The final weights using the various approximations are almost the same. Using a linear approximation resulted

CONVERGENCE HISTORY FOR VARIOUS APPROXIMATIONS (FRAME015)

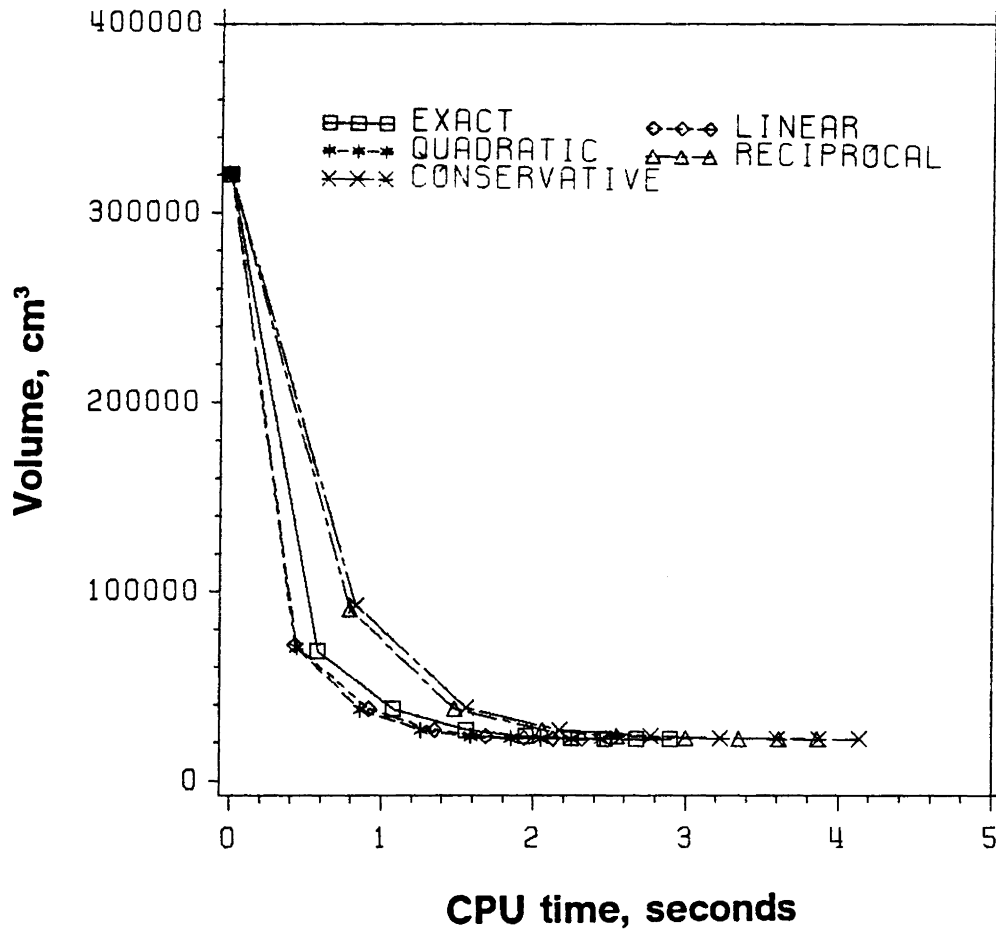


Figure 5. Optimization Convergence History for Various Approximations for Portal Frame FRAME015

in savings of about 75 percent in the number of constraint evaluations, while the savings was even greater for the quadratic, reciprocal and conservative approximations (see Tables 11, 12, 13 14 and 15).

Using the decoupling method similar weights for the final design were obtained for exact constraints (Table 11), and the linear (Table 12), quadratic (Table 13), reciprocal (Table 14) and conservative (Table 15) approximations of the constraints.

Figure 6 shows the variation of the volume of the frame with CPU time for exact constraints. In each response surface the same design is obtained from using either the coupled or the decoupled system with a slight increase in computational time. CPU times are summarized in Figure 7. There was a 20-30 percent reduction in computational time for the evaluation of the hessian matrix, and a 20-25 percent savings in computational time for the evaluation of constraint derivatives. For this small problem these large percentage savings were not enough to compensate for the increased effort required using the implementation of the decoupling algorithm included in "OTHER". This resulted in a 3-7 percent increase in the total CPU time required for the optimization. Figure 8 shows a 30-50 percent reduction in the number of times a global analysis was required. Each constraint derivative evaluation requires 16 global analyses for constraint evaluations for the coupled system, while only 7 global analyses are necessary for constraint evaluations for the decoupled system. The use of approximations can reduce the number of

Table 11. Optimization Results for Portal Frame FRAME015 using Exact Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	22002.242	22002.222	
Objective function, cm ³	21988.049	21988.169	
Number of ODMs	36	37	
Global analyses	993	697	29.81
Number of Evaluations			
Objective function	1532	1584	
Constraints	416	437	
Constraint gradients	36	37	
Approximate constraints	0	0	
CPU Times *			
Total	2.91	3.13	-7.56
ODM	0.86	0.84	
Direction	1.99	2.18	
Hessian setup	0.76	0.53	30.26
Solution	0.07	0.10	
Objective function	0.14	0.13	
Constraints	0.65	0.60	
Constraint gradients	0.85	0.68	20.00
Approximate constraints	0.00	0.00	
Area 1	13.266168	13.265738	
Area 2	13.878708	13.879007	
Area 3	1.476257	1.476293	

* CPU seconds on IBM 3084

Table 12. Optimization Results for Portal Frame FRAME015 using Linear Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	22002.192	22002.235	
Objective function, cm ³	21988.085	21988.150	
Number of ODMs	36	37	
Global analyses	687	364	47.02
Number of Evaluations			
Objective function	1642	1667	
Constraints	110	104	
Constraint gradients	36	37	
Approximate constraints	416	416	
CPU Times *			
Total	2.51	2.69	-7.17
ODM	0.49	0.47	
Direction	1.94	2.13	
Hessian setup	0.74	0.57	22.97
Solution	0.08	0.07	
Objective function	0.19	0.21	
Constraints	0.16	0.18	
Constraint gradients	0.85	0.63	25.88
Approximate constraints	0.06	0.01	
Area 1	13.266218	13.265797	
Area 2	13.878695	13.878967	
Area 3	1.476281	1.476284	

* CPU seconds on IBM 3084

Table 13. Optimization Results for Portal Frame FRAME015 using Quadratic Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	22002.237	22002.227	
Objective function, cm ³	21987.960	21988.265	
Number of ODMs	36	36	
Global analyses	650	326	49.85
Number of Evaluations			
Objective function	1537	1529	
Constraints	73	73	
Constraint gradients	36	36	
Approximate constraints	348	340	
CPU Times *			
Total	2.45	2.56	-4.49
ODM	0.43	0.36	
Direction	1.93	2.10	
Hessian setup	0.75	0.50	33.33
Solution	0.05	0.12	
Objective function	0.13	0.12	
Constraints	0.14	0.12	
Constraint gradients	0.83	0.64	22.89
Approximate constraints	0.05	0.05	
Area 1	13.266039	13.265678	
Area 2	13.878744	13.879091	
Area 3	1.476196	1.476334	

* CPU seconds on IBM 3084

Table 14. Optimization Results for Portal Frame FRAME015 using Reciprocal Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	22002.337	22002.342	
Objective function, cm ³	21988.306	21988.345	
Number of ODMs	38	36	
Global analyses	648	306	52.78
Number of Evaluations			
Objective function	1607	1607	
Constraints	39	39	
Constraint gradients	38	38	
Approximate constraints	390	390	
CPU Times *			
Total	3.87	4.01	-3.62
ODM	1.66	1.70	
Direction	2.13	2.23	
Hessian setup	0.80	0.54	30.50
Solution	0.09	0.09	
Objective function	0.15	0.17	
Constraints	0.08	0.08	
Constraint gradients	0.90	0.67	25.56
Approximate constraints	1.36	1.46	
Area 1	13.266181	13.265700	
Area 2	13.878907	13.879170	
Area 3	1.476309	1.476325	

* CPU seconds on IBM 3084

Table 15. Optimization Results for Portal Frame FRAME015 using Conservative Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	22002.353	22002.357	
Objective function, cm ³	21988.346	21988.339	
Number of ODMs	39	39	
Global analyses	665	314	52.78
Number of Evaluations			
Objective function	1648	1648	
Constraints	40	40	
Constraint gradients	39	39	
Approximate constraints	399	399	
CPU Times *			
Total	4.14	4.28	-3.38
ODM	1.91	1.89	
Direction	2.14	2.29	
Hessian setup	0.81	0.57	29.63
Solution	0.04	0.08	
Objective function	0.19	0.16	
Constraints	0.08	0.04	
Constraint gradients	0.91	0.71	21.98
Approximate constraints	1.60	1.70	
Area 1	13.266188	13.265700	
Area 2	13.878939	13.879170	
Area 3	1.476313	1.476325	

* CPU seconds on IBM 3084

global analyses required substantially and there is a further 50 percent reduction if the decoupled system is used.

6.2 Numerical Results for Portal Frame FRAME050

The portal frame was next modeled with a finer mesh, with each element being 250 cm long. Thus there were 2 elements for beam 1, and 4 elements each for beams 2 and 3 for a total of 10 frame elements (see Figure 9). Each element had 5 design variables controlling its cross sectional shape. Thus in this model, denoted FRAME050 there were a total of 50 design variables, including 20 global variables. The total number of constraints was 248, 8 global displacement constraints and 24 constraints on stresses and upper and lower bounds in each element.

Figure 10 shows the variation of the volume of the frame as a function of the CPU time used during optimization for exact constraints, and the linear, quadratic, reciprocal and conservative approximations of the constraints using the original coupled system. Again, the reciprocal and the conservative approximations take substantially longer to converge to the weight obtained by the other methods, but the final weights for all approximations are almost the same. Using a linear approximation resulted in a savings of about 85 percent in the number of constraint evaluations, while the savings were even greater for the quadratic, reciprocal and conservative approximations (see Tables 16, 17, 18, 19 and 20).

CONVERGENCE HISTORY FOR COUPLED AND DECOUPLED SYSTEM (FRAME015)

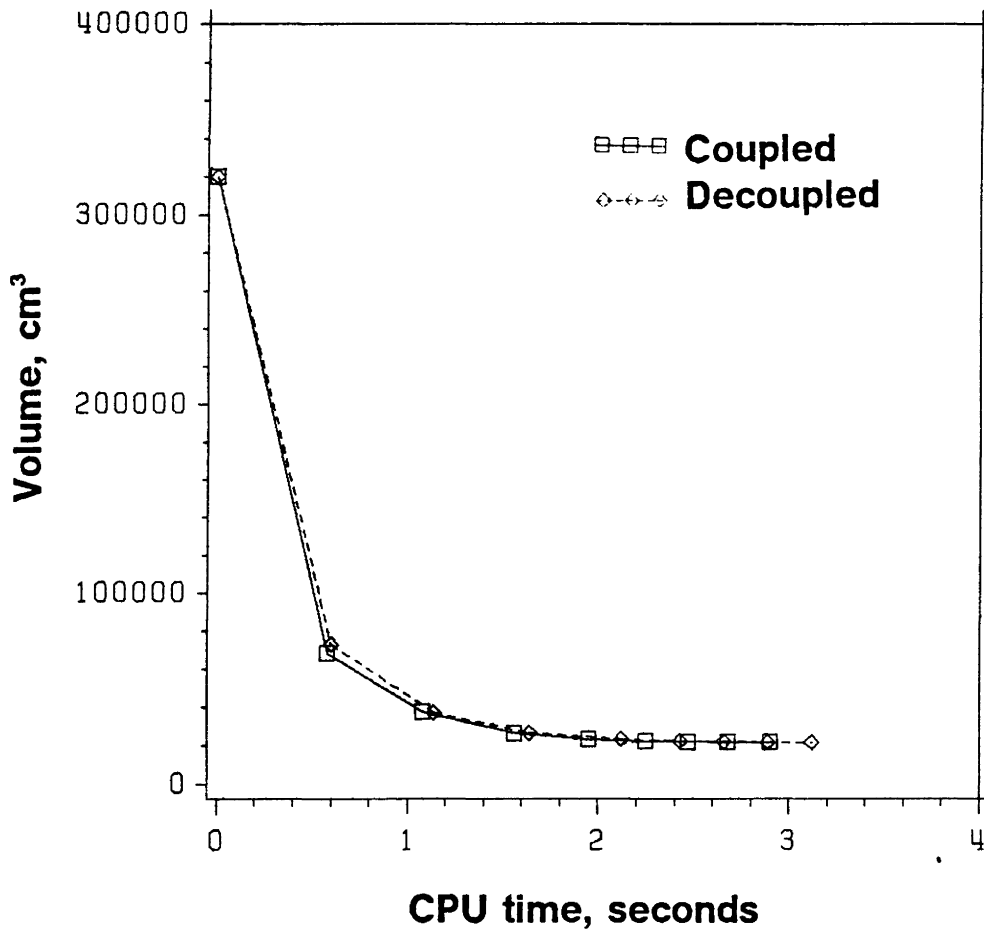


Figure 6. Optimization Convergence History for Coupled and Decoupled Systems with Exact Constraints for Portal Frame FRAME015

CPU TIMES WITH EXACT AND LINEAR APPROXIMATIONS (FRAME015)

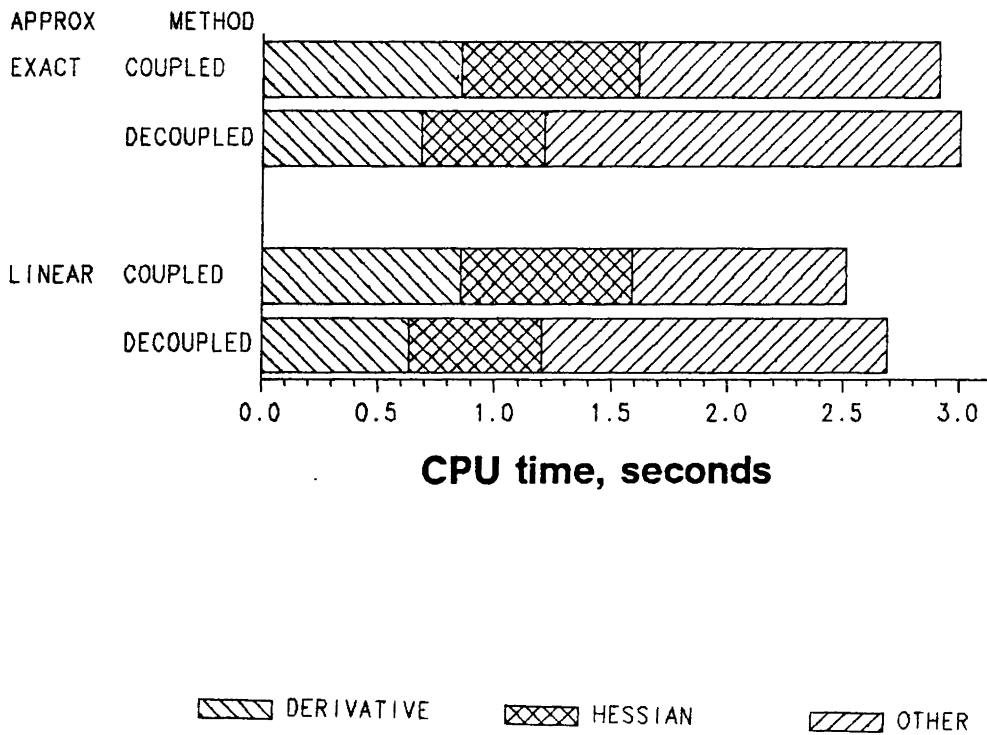


Figure 7. CPU Times for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME015

NUMBER OF GLOBAL ANALYSES WITH EXACT AND LINEAR APPROXIMATIONS (FRAME015)

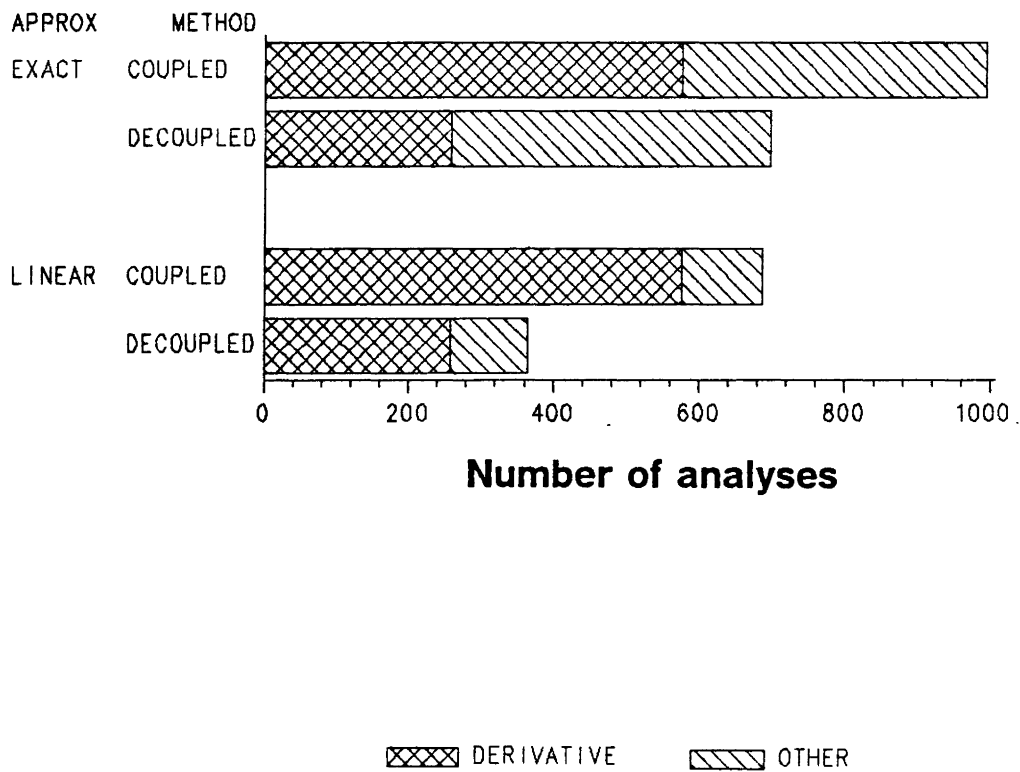
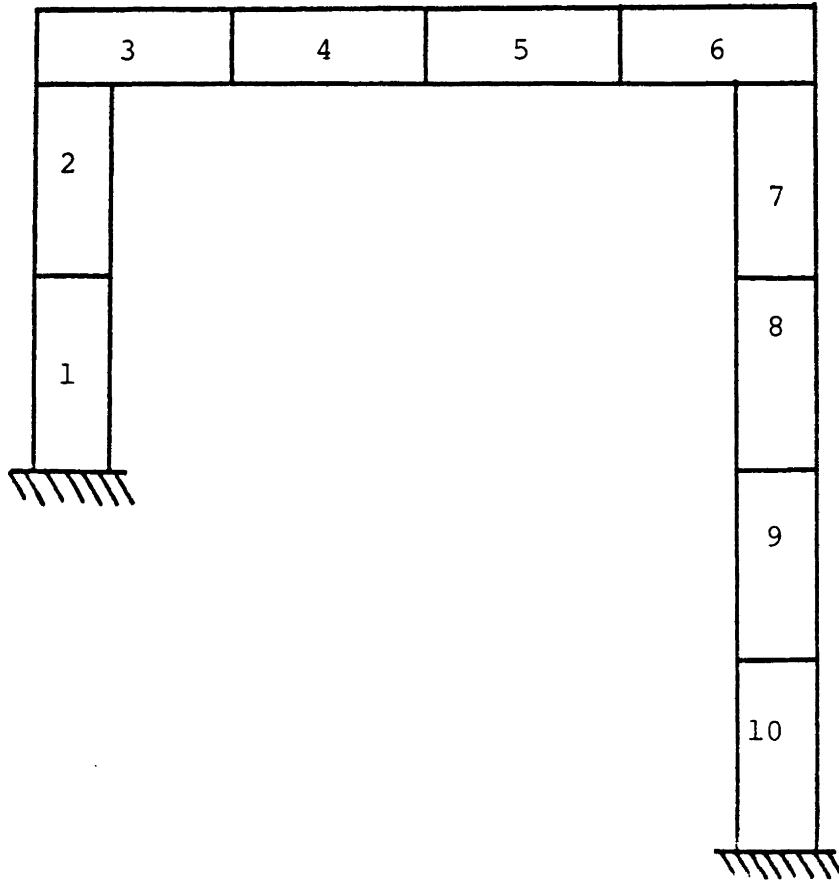


Figure 8. Number of Global Analyses for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME015

FINITE ELEMENT MESH FOR PORTAL FRAME (FRAME050)



Dimension	Beam 1	Beam i
d_1	X_1	X_{5i-4}
d_2	X_2	X_{5i-3}
d_3	X_3	X_{5i-2}
d_4	X_4	X_{5i-1}
d_5	5.000 cm	5.000 cm
d_6	X_5	X_{5i}

Figure 8. Finite element model for portal frame FRAME50

CONVERGENCE HISTORY FOR VARIOUS APPROXIMATIONS (FRAME050)

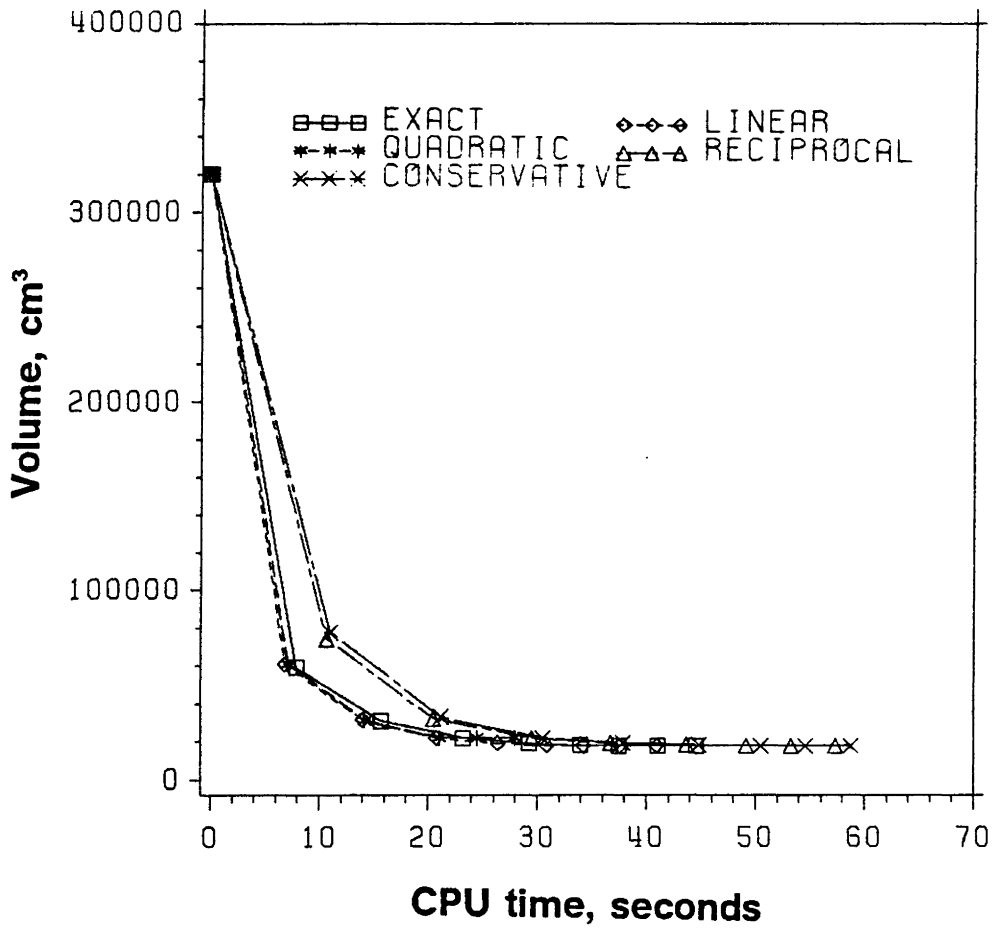


Figure 10. Optimization Convergence History for Various Approximations for Portal Frame FRAME050

Table 16. Optimization Results for Portal Frame FRAME050 using Exact Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	18057.518	18057.508	
Objective function, cm ³	18045.187	18045.184	
Number of ODMs	36	36	
Global analyses	2296	1193	48.04
Number of Evaluations			
Objective function	4095	4072	
Constraints	459	436	
Constraint gradients	36	36	
Approximate constraints	0	0	
CPU Times *			
Total	44.56	31.51	29.29
ODM	4.87	4.65	
Direction	39.51	26.71	
Hessian setup	18.99	8.98	52.72
Solution	1.60	1.57	
Objective function	1.21	1.17	
Constraints	4.21	4.01	
Constraint gradients	16.89	9.35	44.64
Approximate constraints	0.00	0.00	

* CPU seconds on IBM 3084

Table 17. Optimization Results for Portal Frame FRAME050 using Linear Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	18057.501	18057.501	
Objective function, cm ³	18045.193	18045.191	
Number of ODMs	36	36	
Global analyses	1910	830	56.54
Number of Evaluations			
Objective function	4119	4123	
Constraints	73	73	
Constraint gradients	36	36	
Approximate constraints	410	414	
CPU Times *			
Total	40.95	28.19	31.16
ODM	1.42	1.41	
Direction	39.36	26.59	
Hessian setup	18.92	8.97	52.59
Solution	1.56	1.55	
Objective function	1.12	1.16	
Constraints	0.70	0.72	
Constraint gradients	16.86	9.31	44.78
Approximate constraints	0.09	0.09	

* CPU seconds on IBM 3084

Table 18. Optimization Results for Portal Frame FRAME050 using Quadratic Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	22321.284	32032.577	
Objective function, cm ³	22221.788	31295.563	
Number of ODMs	24	18	
Global analyses	1310	452	65.50
Number of Evaluations			
Objective function	2903	2230	
Constraints	85	73	
Constraint gradients	24	18	
Approximate constraints	394	339	
CPU Times *			
Total	28.01	14.73	47.41
ODM	1.59	1.35	
Direction	26.28	13.26	
Hessian setup	12.58	4.47	64.47
Solution	1.11	0.79	
Objective function	0.76	0.62	
Constraints	0.79	0.68	
Constraint gradients	11.25	4.63	58.84
Approximate constraints	0.18	0.15	

* CPU seconds on IBM 3084

Table 19. Optimization Results for Portal Frame FRAME050 using Reciprocal Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	18057.640	18057.631	
Objective function, cm ³	18045.335	18045.340	
Number of ODMs	38	38	
Global analyses	1978	838	57.63
Number of Evaluations			
Objective function	4276	4279	
Constraints	39	39	
Constraint gradients	38	38	
Approximate constraints	399	402	
CPU Times *			
Total	57.35	43.93	23.40
ODM	15.51	15.72	
Direction	41.65	28.00	
Hessian setup	20.04	9.42	52.99
Solution	1.64	1.63	
Objective function	1.24	1.21	
Constraints	0.38	0.37	
Constraint gradients	17.78	9.79	44.94
Approximate constraints	14.61	14.77	

* CPU seconds on IBM 3084

Table 20. Optimization Results for Portal Frame FRAME050 using Conservative Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	18057.699	18057.692	
Objective function, cm ³	18045.393	18045.400	
Number of ODMs	38	38	
Global analyses	1978	838	57.63
Number of Evaluations			
Objective function	4279	4283	
Constraints	39	39	
Constraint gradients	38	38	
Approximate constraints	402	406	
CPU Times *			
Total	58.70	45.42	22.62
ODM	17.02	17.30	
Direction	41.49	27.96	
Hessian setup	19.97	9.42	52.83
Solution	1.63	1.65	
Objective function	1.17	1.27	
Constraints	0.39	0.36	
Constraint gradients	17.75	9.73	45.18
Approximate constraints	16.12	16.35	

* CPU seconds on IBM 3084

CONVERGENCE HISTORY FOR COUPLED AND DECOUPLED SYSTEM (FRAME050)

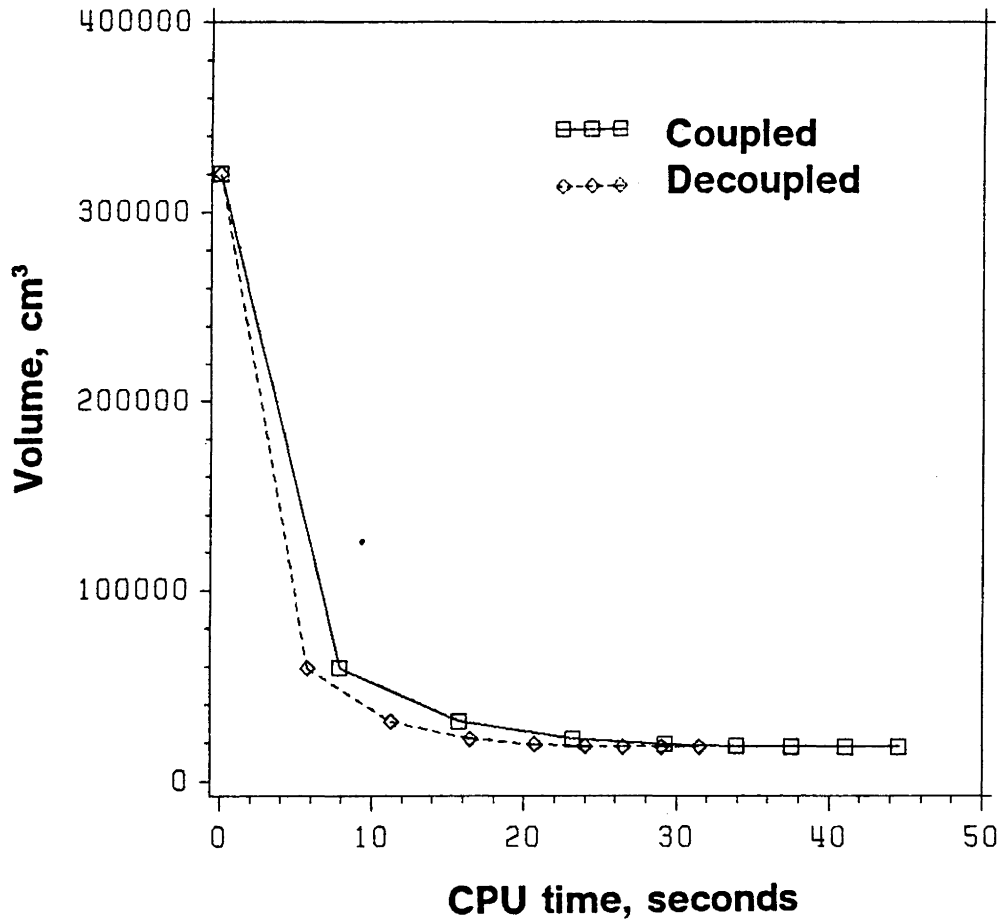


Figure 11. Optimization Convergence History for Coupled and Decoupled Systems with Exact Constraints for Portal Frame FRAME050

CPU TIMES WITH EXACT AND LINEAR APPROXIMATIONS (FRAME050)

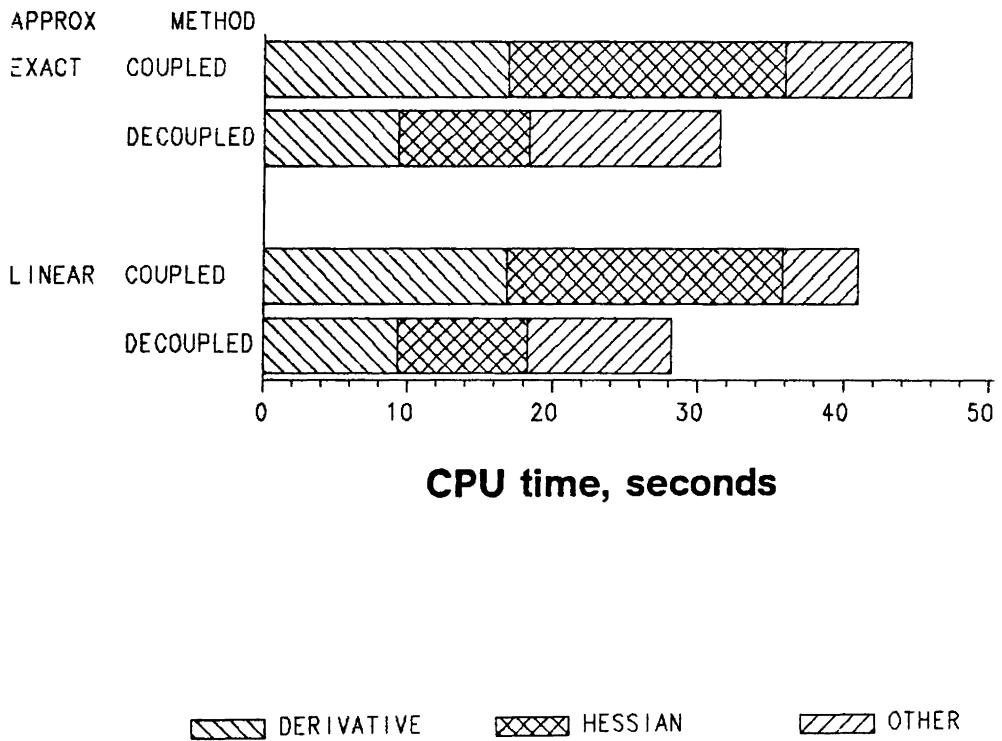


Figure 12. CPU Times for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME050

**NUMBER OF GLOBAL ANALYSES
WITH EXACT AND LINEAR APPROXIMATIONS (FRAME050)**

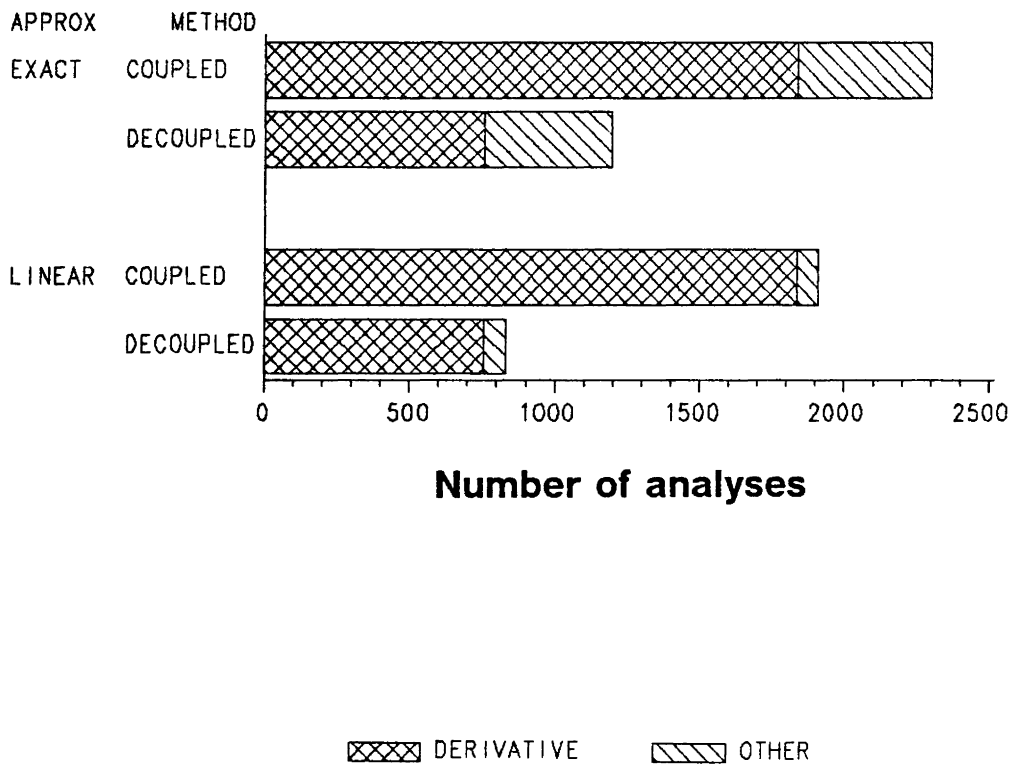


Figure 13. Number of Global Analyses for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME050

Using the decoupling scheme similar weights for the final design were obtained for exact constraints (Table 16), and the linear (Table 17), reciprocal (Table 19) and conservative (Table 20) approximations of the constraints. The quadratic approximation for the constraints did not converge to the optimum value obtained by the other methods even for the coupled system (Table 18). This could be due to the fact that the quadratic approximation has a larger error than the linear approximation at distances further away from the base point. It may be useful to use move-limits to guard against such deterioration in quality of the quadratic approximation. Figure 11 shows the variation of the volume of the frame with CPU time for exact constraints. In each response surface the same design is obtained from using either the coupled or the decoupled system with a 20-30 percent reduction in computational time. A summary of the CPU times for this case is shown in Figure 12. There was a 50 percent reduction in computational time for the evaluation of the hessian matrix, and a 45-60 percent savings in computational time for the evaluation of constraint derivatives. The savings from these was much larger than the increased effort required using the implementation of the decoupling algorithm. This resulted in a 20-30 percent reduction in the total CPU time required for the optimization. There was a 45-60 percent reduction in the number of times a global analysis was required (see Figure 13). Each constraint derivative evaluation requires 51 global analyses for constraint evaluations for the coupled system, while only 21 global analyses are necessary for constraint evaluations for the decoupled system. The use of

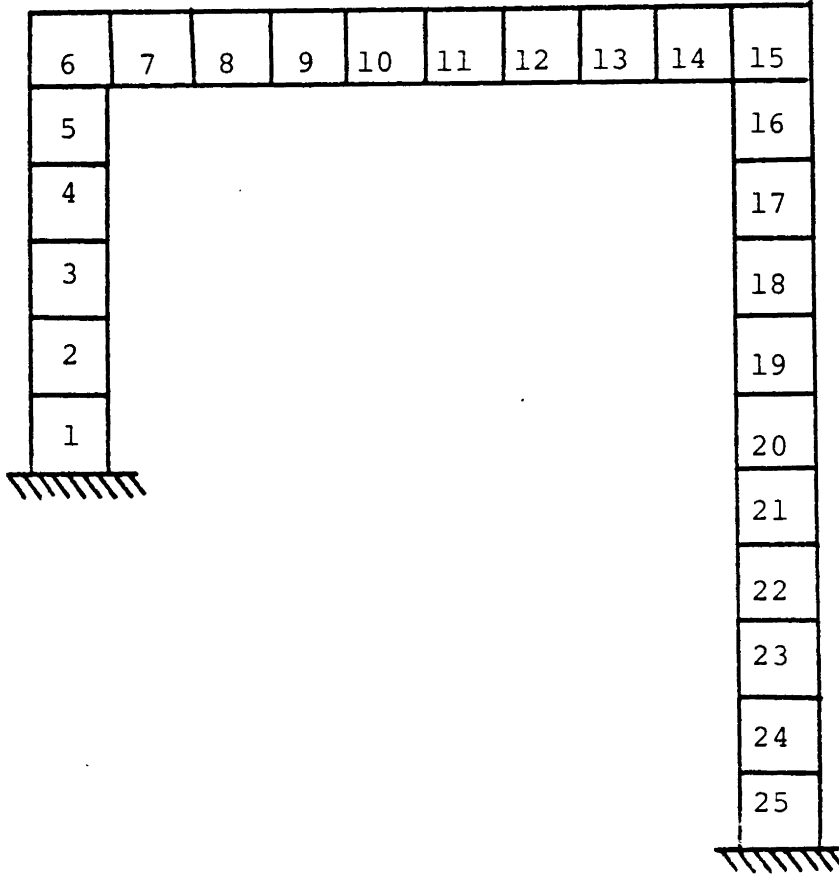
approximations can reduce the number of global analyses required substantially and there is a further 60 percent reduction if the decoupled system is used.

6.3 Numerical Results for Portal Frame FRAME125

Next, an even finer mesh was used to model the portal frame with 100 cm long elements. Thus there were 5 elements for beam 1 and 10 elements for each of beams 2 and 3 for a total of 25 frame elements (see Figure 14), and a total of 125 design variables. This model was named FRAME125. The number of global variables was 50, and the total number of constraints was 608, 8 global displacement constraints and 24 constraints on stresses and upper and lower bounds in each element.

Figure 15 shows the variation of the volume of the frame as a function of the CPU time used during optimization for exact constraints and the linear approximations of the constraints using the original coupled system. For this model the quadratic, reciprocal, and conservative approximations were not used since they had shown convergence problems for the previous model FRAME050. The final weight using the linear approximation is almost identically the same as that obtained from using exact constraints. Using a linear approximation resulted in a savings of about 85 percent in the number of constraint evaluations (see Tables 21 and 22).

FINITE ELEMENT MESH FOR PORTAL FRAME (FRAME125)



Dimension	Beam 1	Beam i
d_1	X_1	X_{5i-4}
d_2	X_2	X_{5i-3}
d_3	X_3	X_{5i-2}
d_4	X_4	X_{5i-1}
d_5	5.000 cm	5.000 cm
d_6	X_5	X_{5i}

Figure 13. Finite element model for portal frame FRAME125

CONVERGENCE HISTORY FOR EXACT AND LINEAR APPROXIMATIONS (FRAME125)

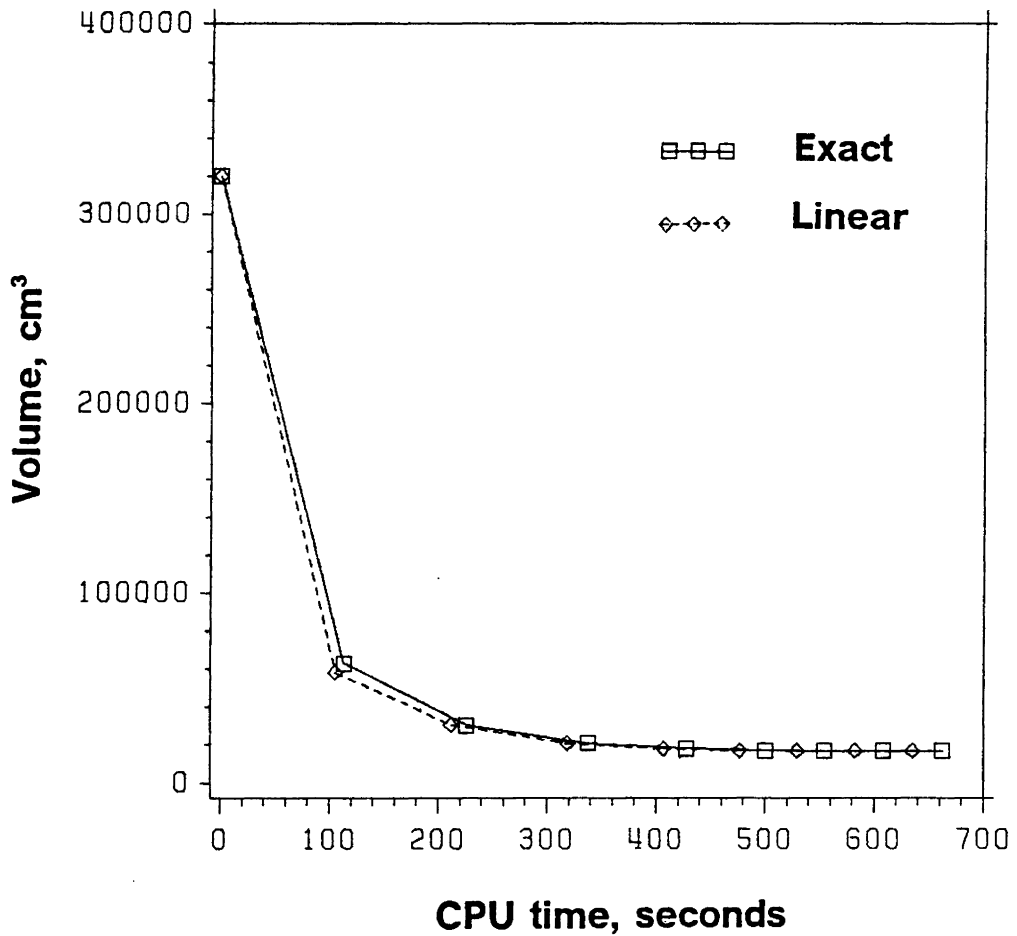


Figure 15. Optimization Convergence History for Exact and Linear Approximation for Constraints for Portal Frame FRAME125

Table 21. Optimization Results for Portal Frame FRAME125 using Exact Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	16772.968	16772.932	
Objective function, cm ³	16761.411	16761.375	
Number of ODMs	36	36	
Global analyses	5023	2319	53.83
Number of Evaluations			
Objective function	9522	9518	
Constraints	486	482	
Constraint gradients	36	36	
Approximate constraints	0	0	
CPU Times *			
Total	662.56	342.48	48.31
ODM	34.94	34.47	
Direction	627.19	307.58	
Hessian setup	278.19	106.51	61.71
Solution	21.96	21.65	
Objective function	6.28	6.37	
Constraints	33.49	33.04	
Constraint gradients	316.06	140.61	55.51
Approximate constraints	0.00	0.00	

* CPU seconds on IBM 3084

Table 22. Optimization Results for Portal Frame FRAME125 using Linear Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, cm ³	16772.972	16772.973	
Objective function, cm ³	16761.406	16761.508	
Number of ODMs	36	36	
Global analyses	4610	1913	58.50
Number of Evaluations			
Objective function	9557	9569	
Constraints	73	76	
Constraint gradients	36	36	
Approximate constraints	448	457	
CPU Times *			
Total	635.16	318.24	49.90
ODM	6.76	7.03	
Direction	627.96	310.72	
Hessian setup	278.83	107.96	61.28
Solution	21.95	22.00	
Objective function	6.26	6.38	
Constraints	5.09	5.31	
Constraint gradients	316.13	141.67	55.19
Approximate constraints	0.25	0.21	

* CPU seconds on IBM 3084

CONVERGENCE HISTORY FOR COUPLED AND DECOUPLED SYSTEM (FRAME125)

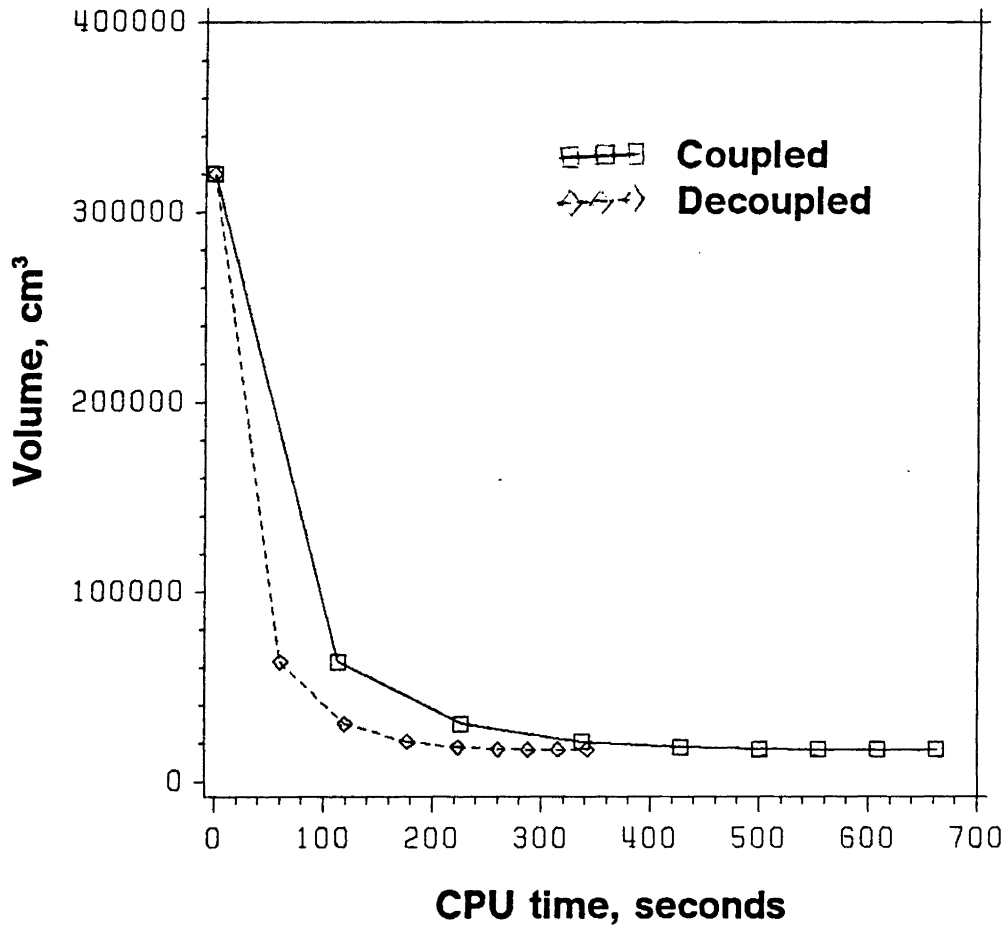


Figure 16. Optimization Convergence History for Coupled and Decoupled Systems with Exact Constraints for Portal Frame FRAME125

CPU TIMES WITH EXACT AND LINEAR APPROXIMATIONS (FRAME125)

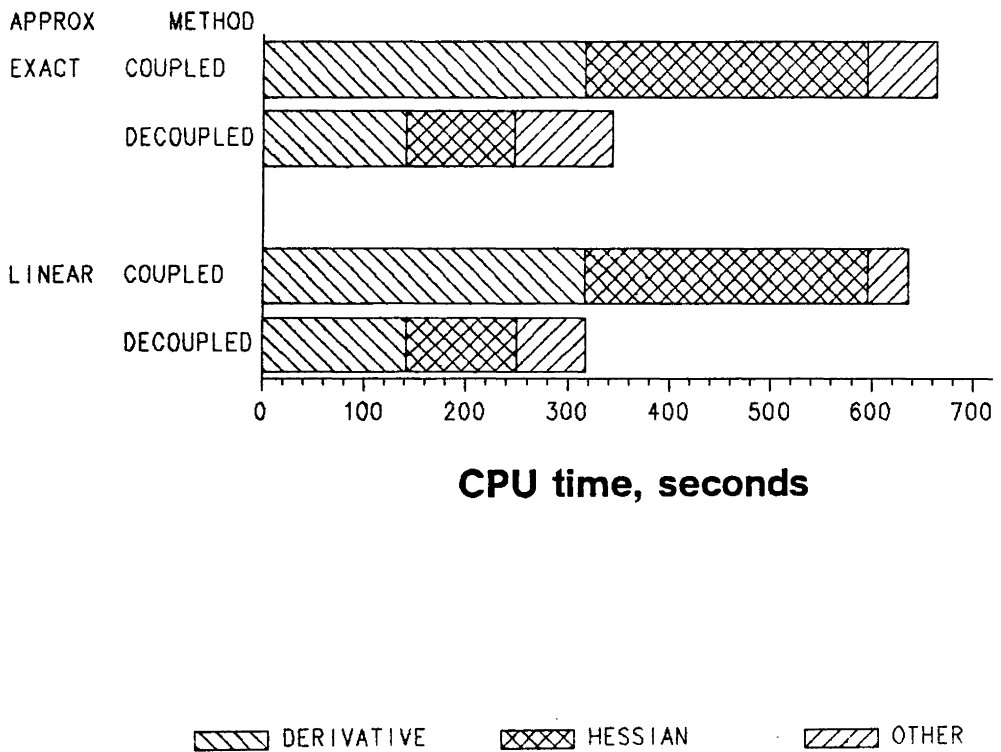


Figure 17. CPU Times for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME125

NUMBER OF GLOBAL ANALYSES WITH EXACT AND LINEAR APPROXIMATIONS (FRAME125)

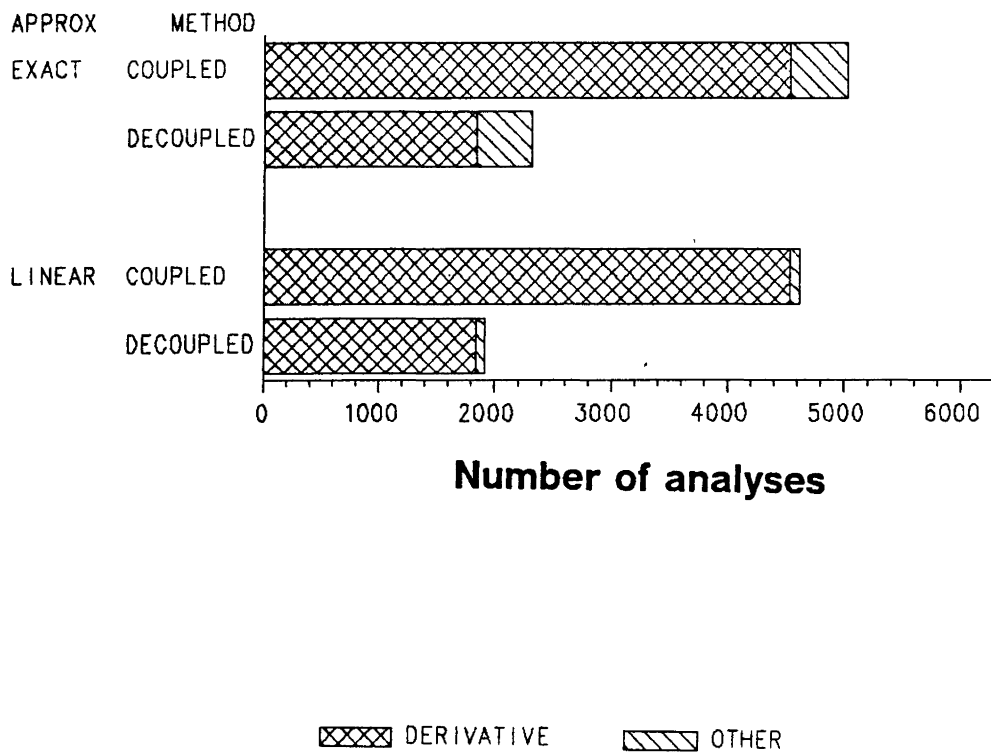


Figure 18. Number of Global Analyses for Coupled and Decoupled Systems with Exact and Linear Approximations for Portal Frame FRAME125

Using the decoupling technique similar weights for the final design were obtained for exact constraints (Table 21) and the linear (Table 22) approximation of the constraints. Figure 16 shows the variation of the volume of the frame with CPU time for exact constraints. In each response surface the same design is obtained from using either the coupled or the decoupled system with almost a 50 percent reduction in computational time. Figure 17 summarizes the CPU times. There was a 60 percent reduction in computational time for the evaluation of the hessian matrix, and a 55 percent savings in computational time for the evaluation of constraint derivatives. For this case too the savings from these was much larger than the increased effort required for the implementation of the decoupling algorithm. This resulted in almost a 50 percent reduction in the total CPU time required for the optimization. There was a 50-60 percent reduction in the number of times a global analysis was required (see Figure 18). Each constraint derivative evaluation requires 126 global analyses for constraint evaluations for the coupled system, while only 51 global analyses are necessary for constraint evaluations for the decoupled system. The use of approximations can reduce the number of global analyses required substantially and there is a further 60 percent reduction if the decoupled system is used.

6.4 Comparison of Results for All Models of the Portal Frame

Let us now look at the impact of increasing the number of elements and design variables used to model the portal frame, as shown for FRAME015, FRAME050 and FRAME125 earlier. A greater number of design variables gives the optimizer greater freedom to put material where it is required and remove it where it is not. As a result of this, the minimum volume decreased from 21,988 cm³ for FRAME015 to 18,045 cm³ for FRAME050 and 16,761 cm³ for FRAME125 (Table 23).

The optimized distribution of cross sectional area for beam 1 and 2 for the three sizes of the element used is shown in Figures 19 and 20 respectively. For all three cases the cross sectional area of beam 3 was at the lower bound.

A summary of the CPU time savings for the three models of the portal frame is shown in Figures 21, 22 and 23. For FRAME015 the savings from the hessian evaluation and the constraints gradients using a decoupled system are not enough to offset the increase in cost of decoupling. But as the size of the model and the amount of sparsity in the hessian matrix increases, greater savings are obtained when the decoupled system is used. The amount of savings resulting from the constraint gradient calculations using global variables instead of local variables is limited to 60 percent for this example since there are 5 local variables and 2 global variables for each subsystem.

Table 23. Comparison of Results for all Models of the Portal Frame

	FRAME015	FRAME050	FRAME125
Number of design variables	15	50	125
Number of constraints	80	248	608
Optimized volume, cm ³	21988	18045	16761
Average cost of a global analysis *	1.481E-3	9.185E-3	6.963E-2
Percent Savings from using Decoupling Technique			
Global analyses	47.02	56.54	58.50
Total CPU time *	-7.17	31.16	49.90
Hessian setup time *	22.97	52.59	61.28
Constraint gradient time *	25.88	44.78	55.19

* CPU seconds on IBM 3084

CROSS-SECTIONAL AREA VARIATION FOR BEAM 1

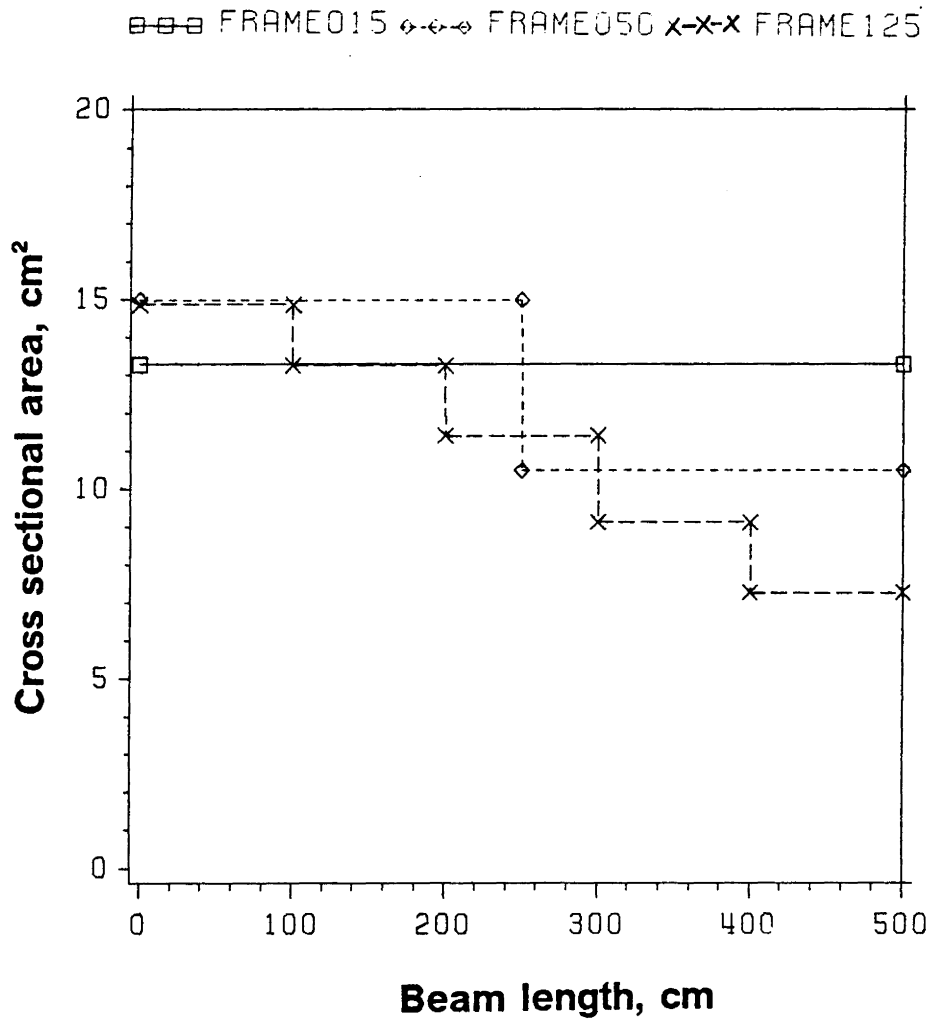


Figure 19. Cross Sectional Area Variation for Beam 1 for all Models of the Portal Frame

CROSS-SECTIONAL AREA VARIATION FOR BEAM 2

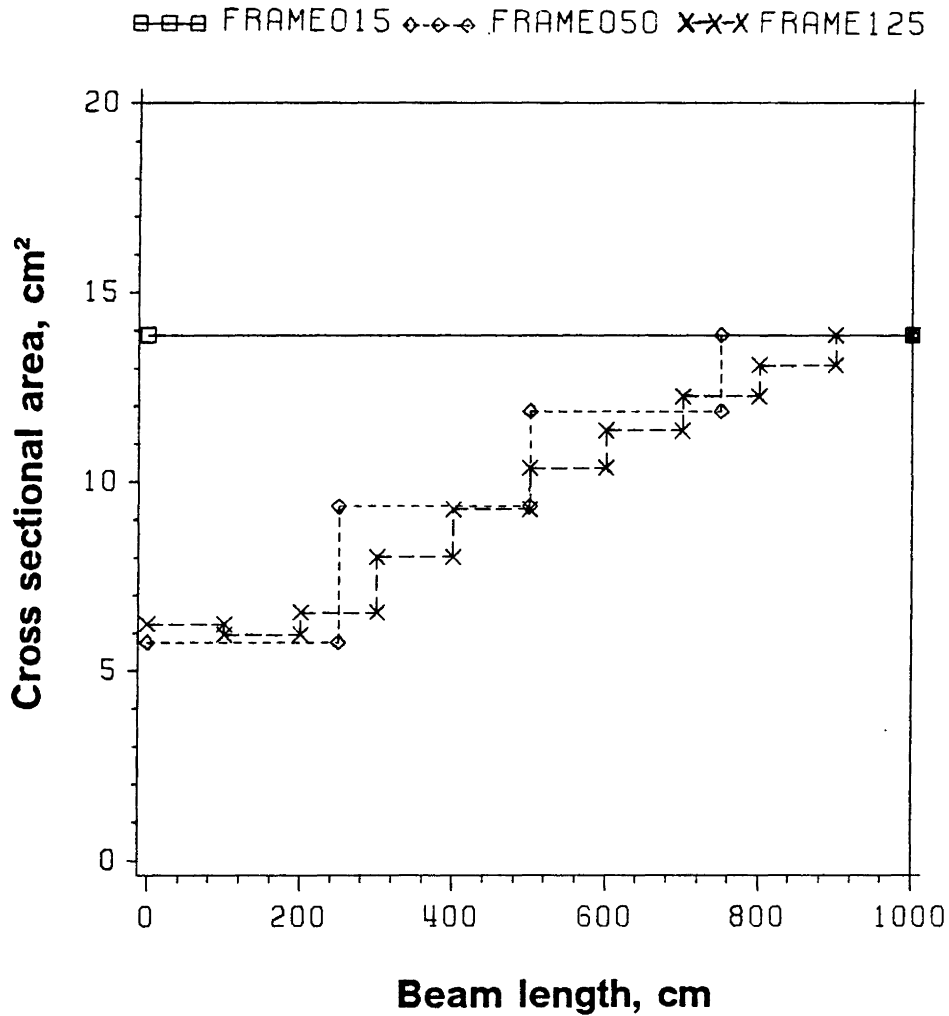


Figure 20. Cross Sectional Area Variation for Beam 2 for all Models of the Portal Frame

For structures where larger number of detailed design variables are used this component of savings will increase proportionately. For very large systems, the evaluation of the hessian matrix can be a dominant portion of the total computational time, and the use of the decoupled system can reduce the computational costs by as much as 80-90 percent.

For the frame example just discussed, the finite element modeling was relatively inexpensive. A global analysis of the largest model FRAME125 required only about 0.07 CPU seconds, while the smallest one FRAME015 needed only about 0.0015 CPU seconds. For complicated structures this time can be significantly larger especially if nonlinearities in the material behavior are considered or a viscoelastic analysis is performed. Using the decoupled system saves 45-60 percent (see Figure 24) in the number of global analyses required (see Table 23).

PERCENT SAVINGS IN TOTAL CPU TIME

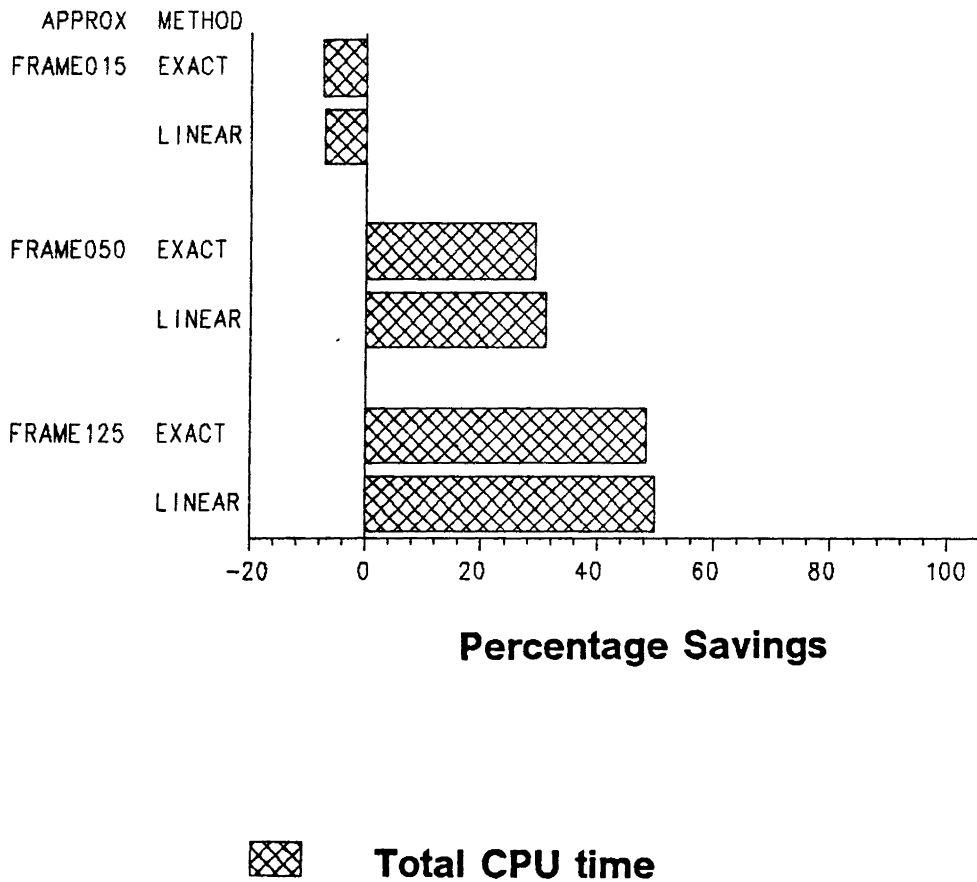


Figure 21. Total CPU Times for Coupled and Decoupled Systems for all Models of the Portal Frame

PERCENT SAVINGS IN HESSIAN SETUP TIME

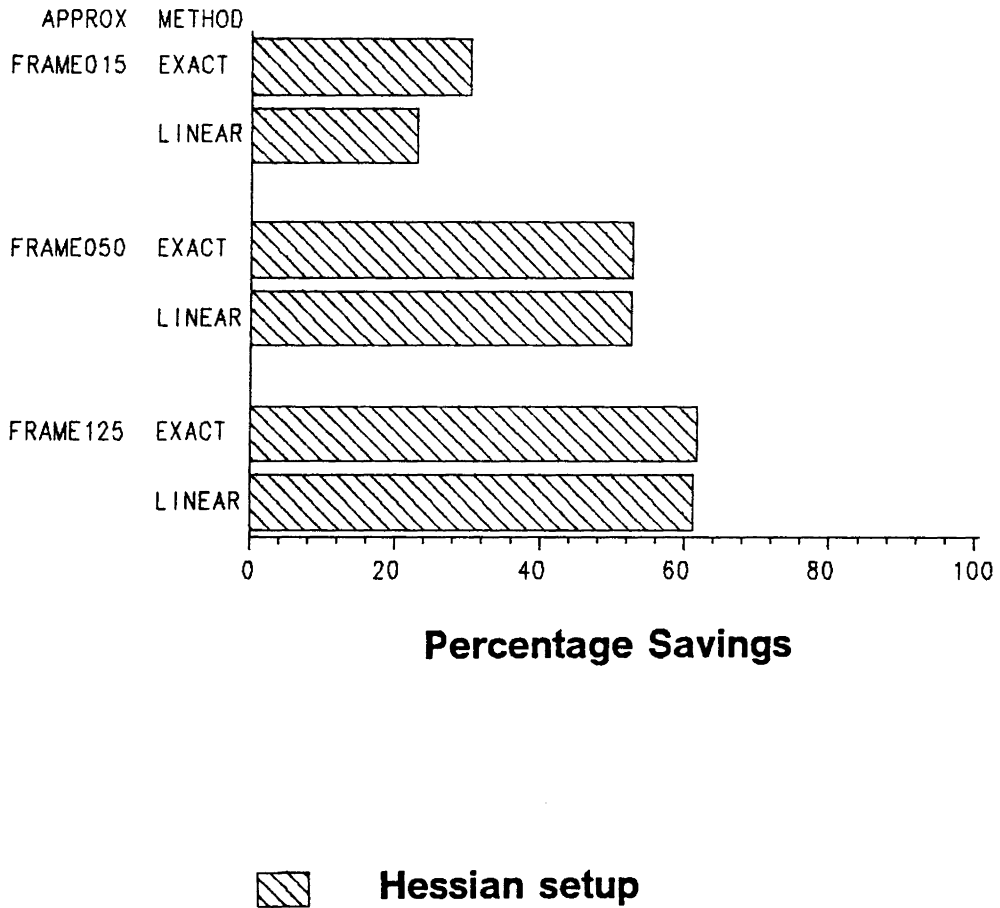


Figure 22. Hessian Setup Time for Coupled and Decoupled Systems for all Models of the Portal Frame

PERCENT SAVINGS IN CONSTRAINT GRADIENT TIME

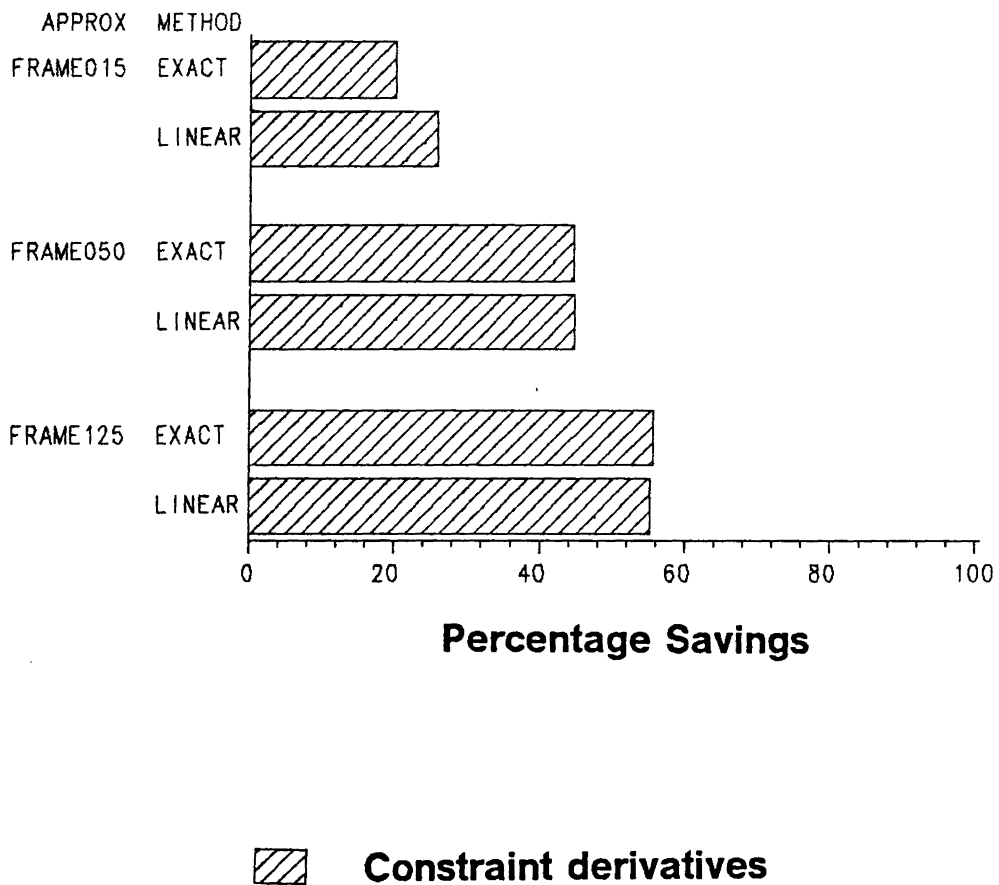


Figure 23. Constraint Gradient Time for Coupled and Decoupled Systems for all Models of the Portal Frame

PERCENT SAVINGS IN NUMBER OF GLOBAL ANALYSES

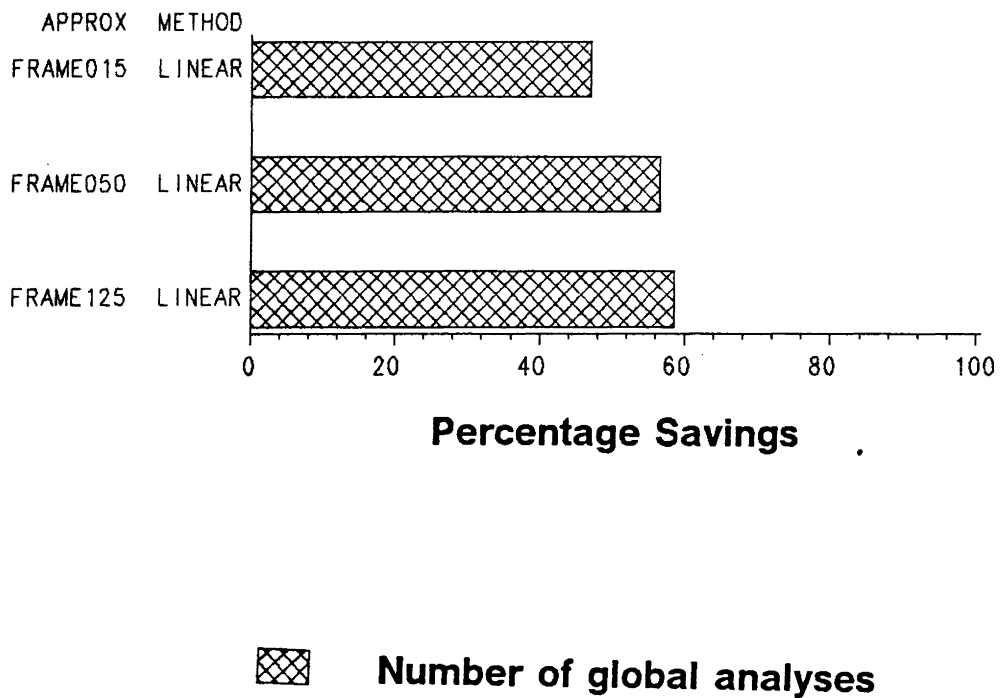


Figure 24. Number of Global Analyses for Coupled and Decoupled Systems for all Models of the Portal Frame

7.0 Numerical Results for Wing Box

In the previous chapter we applied the decoupling scheme to a simple portal frame model. For the frame a global analysis of the structure was simple and cheap. The use of the cross sectional area A and the moment of inertia I as global variables for decoupling has already been investigated (Refs 17, 26, 33 and 35).

7.1 Development of Global Variables for Wing Panels

One of the objectives of this research was to extend the application of the decoupling technique to a simple finite element model of a wing structure (Figure 25). The wing is modeled with truss elements for the spar caps, shear panels for the shear webs and membrane elements (of composite material) for the cover panels. Typically, the cover panels are stiffened sections with

FINITE ELEMENT MODEL FOR WING BOX EXAMPLE

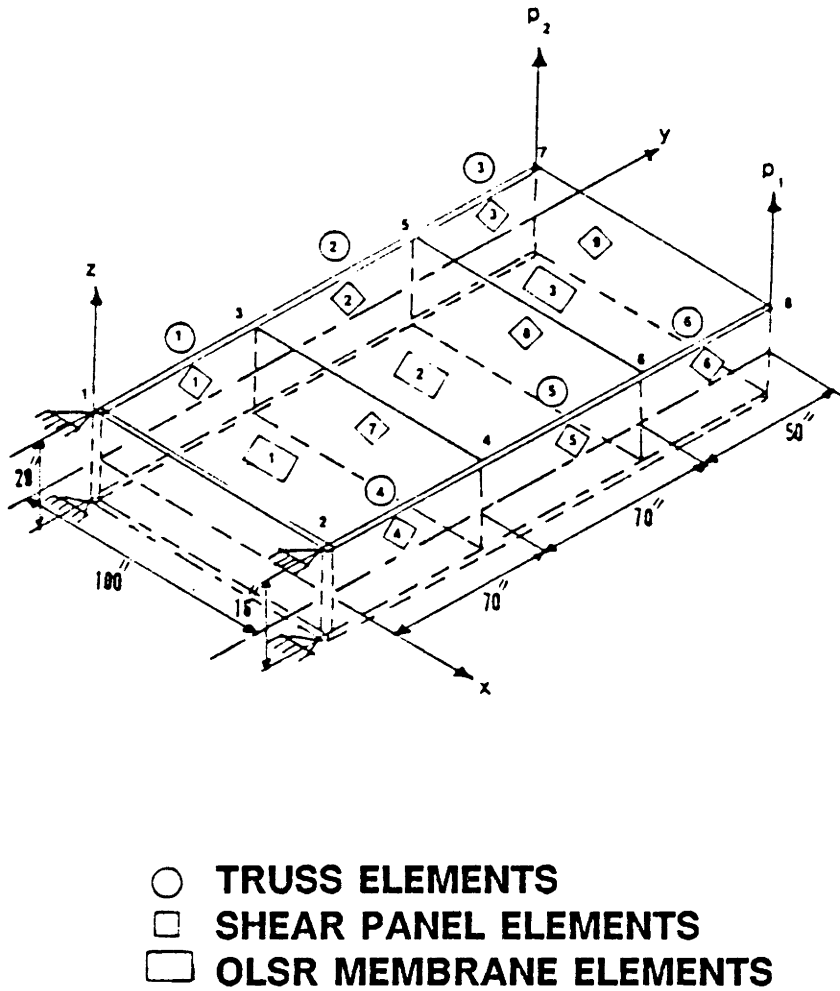


Figure 25. Finite Element Model for Wing Box

complicated cross sections. Each panel on the wing can be considered as an independent substructure. In order to decouple these substructures suitable global quantities need to be defined. With these global quantities fixed, the global response of the wing can be analyzed.

Mehrinfar modeled the stiffness of hat-stiffened wing panels (see Figure 26) at the global level by stacking 5 layers of orthotropic linear stress rectangular (OLSR) membrane elements (Refs 30 and 31). The thicknesses of these elements was adjusted to represent the equivalent thickness of the 0° material, the $+45^\circ$ and -45° material in the backup sheet, and the $+45^\circ$ and -45° material in the stiffener. Three independent thicknesses (t_j) and four independent widths (b_j) defined the cross section defined of each panel (see Figure 26). Here t_2 is the thickness of the $\pm 45^\circ$ material. The thickness of the $\pm 45^\circ$ material in the cap (element 3) and sides (element 2) of the hat-stiffener is the same as the thickness of the $\pm 45^\circ$ material in the backup sheet above the stiffener (element 1). Thus, $t_1(\pm 45^\circ) = t_2(\pm 45^\circ)$. Also, the thickness of the $\pm 45^\circ$ material between the stiffeners (element 4) is twice that of the $\pm 45^\circ$ material in the sides and cap of the stiffener. The thickness of the two equal 0° layer in the cap (element 3) is $t_3(0^\circ)$ while $t_4(0^\circ)$ is the thickness of the 0° layer sandwiched between the $\pm 45^\circ$ skins in element 4. The widths of elements 1, 2 and 3 are b_1 , b_2 and b_3 respectively, while b_4 is the half-width of element 4.

Mehrinfar's development was not general enough for arbitrary composite layups, since it presupposes the existence of 0° and $\pm 45^\circ$ plies in the

DETAILS OF HAT-STIFFENERS FOR COVER PANELS

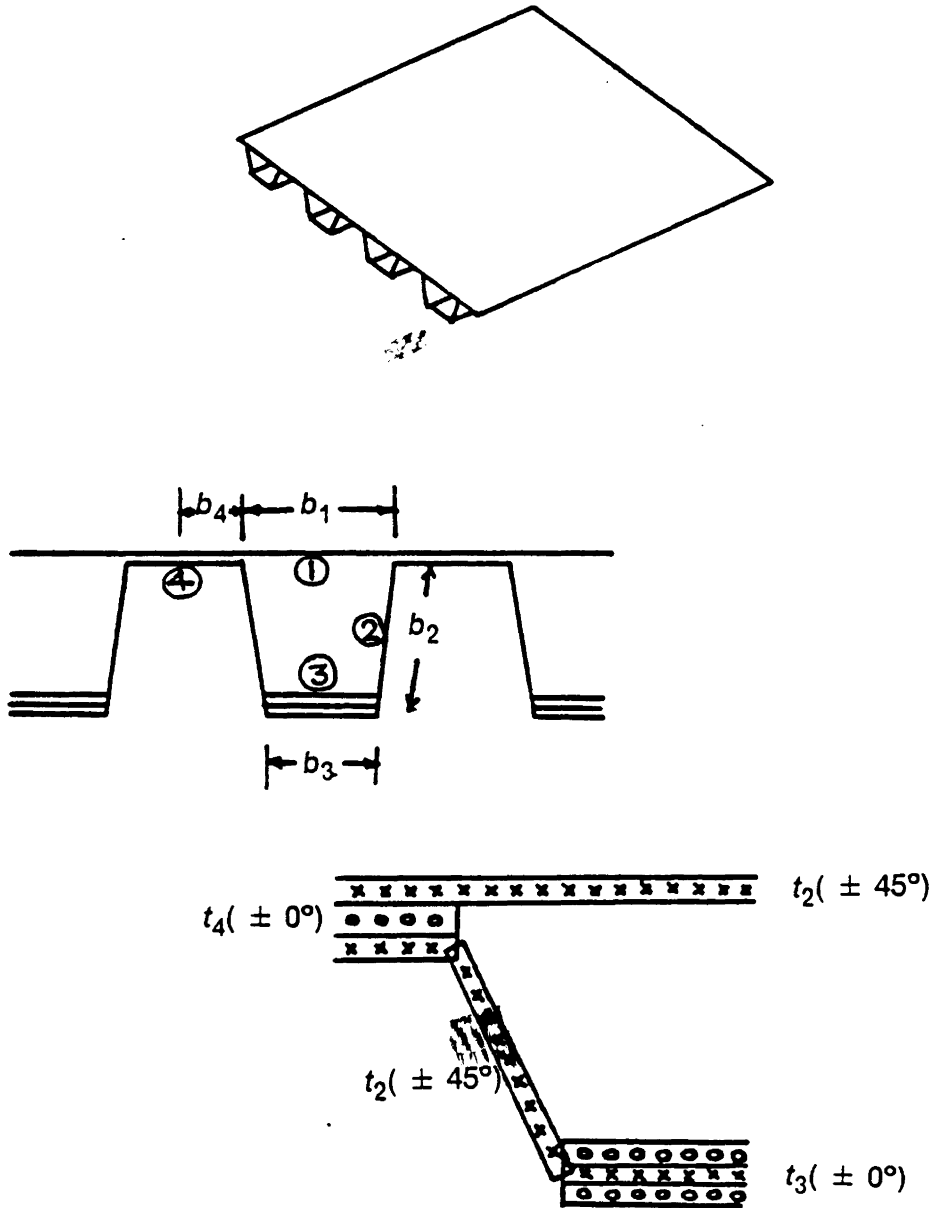


Figure 26. Details of Hat-Stiffeners for the Cover Panels

stiffeners, and the ability of 5 predetermined layers of OLSR membrane elements to correctly model the gross response of the panel.

In this research the use of a single orthotropic membrane element was explored to model the stiffness of the panel at the global level. A general composite stiffened panel was modeled as a single layer of orthotropic material with stiffness properties $A_{11}, A_{12}, A_{22}, A_{66}$ determined from the following relationships of smeared stiffnesses (Ref 31).

$$A_{11} = C_{11\pm 45^\circ} \left[1.0 + \frac{2b_4 + 2b_2 + b_3}{b_1 + 2b_4} \right] t_2(\pm 45^\circ) + C_{110^\circ} \left[\frac{2b_3}{b_1 + 2b_4} \right] t_3(\pm 0^\circ) + C_{110^\circ} \left[\frac{2b_4}{b_1 + 2b_4} \right] t_4(\pm 0^\circ) \quad [7.1]$$

$$A_{12} = C_{12\pm 45^\circ} t_2(\pm 45^\circ) \quad [7.2]$$

$$A_{22} = C_{22\pm 45^\circ} t_2(\pm 45^\circ) \quad [7.3]$$

$$A_{66} = C_{66\pm 45^\circ} \left[1.0 + \frac{2b_4 + 2b_2 + b_3}{b_1 + 2b_4} \right] t_2(\pm 45^\circ) \quad [7.4]$$

where C_{rs0° and $C_{rs\pm 45^\circ}$ are the stiffness coefficients depending on the material properties of the 0° and $\pm 45^\circ$ layers.

Defining the quantities

$$Area = t_2(b_1 + 2b_2 + b_3 + 4b_4) + 2t_3b_3 + 2t_4b_4 \quad [7.5]$$

$$t_{glob} = \frac{Area}{b_1 + 2b_4} \quad [7.6]$$

Then the material properties of the orthotropic layer are defined in terms of the terms of the matrix Q relating membrane strains $\varepsilon_1, \varepsilon_2$ and ε_{12} to membrane stress resultants N_x, N_y and N_{xy} .

$$Q_{11} = A_{11} t_{glob} / 0.1 \quad [7.7]$$

$$Q_{12} = A_{12} t_{glob} / 0.1 \quad [7.8]$$

$$Q_{22} = A_{22} t_{glob} / 0.1 \quad [7.9]$$

$$Q_{66} = A_{66} t_{glob} / 0.1 \quad [7.10]$$

In these expressions Q_{12} is linearly dependent on Q_{22} . The global variables are the stiffness of the orthotropic membrane layer, or the product of Q and the thickness of the composite layup. Thus, three independent material properties Q_{11}, Q_{22}, Q_{66} are treated as global variables, and the thickness of the panel is fixed at a constant value (0.1 cm in this case) and the material properties are scaled proportionately (see equations [7.7] - [7.10] above).

Instead of stress constraints in each of the OLSR layers modeled by Mehrinfar, a maximum allowable strain constraint of 0.002 was used for $\varepsilon_1, \varepsilon_2$ and ε_{12} .

7.2 Description of Wing Box

Mehrinfar and Schmit (Refs 30 and 31) analyzed a wing box shown in Figure 25. Only the upper half of the wing box was modeled due to symmetry about the wing midplane. Two loading conditions were considered at the wing-tips (Table 24). Global displacement constraints were that the z displacement of all free nodes should remain between -3 and 3 inches. The box structure itself consisted of spar caps modeled by 6 truss elements (TRUSS), and shear panels (SSP) modeled by 9 rectangular shear elements. The truss and shear panel elements were considered to be isotropic with material properties given in Table 24 . The cover plate had 3 fiber composite hat-stiffened panels.

Mehrinfar modeled the stiffness of the panels at the global level by stacking 5 layers of orthotropic linear stress rectangular (OLSR) membrane elements (Refs 30 and 31). The thicknesses of these elements was adjusted to represent the equivalent thickness of the 0° material, the $+45^\circ$ and -45° material in the backup sheet, and the $+45^\circ$ and -45° material in the stiffener.

For the global stiffness variables the three independent material properties Q_{11} , Q_{22} , Q_{66} are used for decoupling. Therefore, the 7 original geometric detailed variables are modeled by just 3 global variables.

The weight of the wing box was minimized subject to stress constraints in the truss and shear web elements and strain constraints in the three panel OLSR elements, and global displacement constraints on the z displacements

Table 24. Wings Box Example - Design Information

Isotropic Material Properties for TRUSS & SSP Elements

E	:	10.0E6 psi
ρ	:	0.1 lb/in ³
μ	:	0.3
$\sigma_{\max}^t = \sigma_{\max}^c$:	16000 psi

Composite Material Properties for OLSR Elements

E_1	:	21.0E6 psi
E_2	:	1.70E6 psi
G_{12}	:	0.65E6 psi
ρ	:	0.056 lb/in ³
μ	:	0.21
$F_{1\max}^t$:	1.8E5 psi
$F_{1\max}^c$:	- 1.8E5 psi
$F_{2\max}^t$:	8.0E3 psi
$F_{2\max}^c$:	- 3.0E4 psi
$F_{12\max}$:	1.2E4 psi
$\epsilon_{allowable}$:	0.002

Load Case	Load P_1 (lb)	Load P_2 (lb)
1	0	15000
2	7500	0

of the free nodes. The design variables were the cross sectional areas of the TRUSS elements (T_j), the thicknesses of the SSP elements (S_j) and the detailed dimensions of the hat-stiffener $b_1 - b_4$ and $t_2 - t_4$. The lower bounds constraints on the TRUSS, SSP and OLSR elements are given in Table 25.

7.3 Numerical Results for Wing Box

For the wing box with TRUSS, SSP and OLSR elements all the constraints global constraints since a stress or buckling analysis at the local level was not performed. There were 36 design variables and 105 constraints. The savings from the introduction of global variables and the elimination of an equal number of local variables was significantly diluted by the presence of the 15 global variables in the coupled system (6 truss cross sectional areas and 9 shear panel thicknesses). Further, this problem is not large enough for the savings from the hessian setup to be significant, especially since there were no local constraints. A global finite element analysis of the wing box using WIDOWAC (Ref 79) was much more expensive than that for the portal frame example FRAME125. Constraint gradient evaluation by finite differences, therefore, constituted a dominant portion of the total CPU time (see Tables 26, 27 and 28). Similar final weights are obtained using exact constraints (Table 26), linear (Table 27) and quadratic (Table 28) approximations of the constraints. Though the convergence of these three approximations is very

Table 25. Wings Box Example - Design Variable Information

Dimension Variable	Design Variable	Initial Value	Lower Bound	Upper Bound
$T_j(in^2)$	x_j	0.98	0.1	-
$S_j(in)$	x_{6+i}	0.2	0.02	-
$b_{1j}(in)$	$x_{15+7(i-1)+1}$	4.0	0.8	10.0
$b_{2j}(in)$	$x_{15+7(i-1)+2}$	4.0	0.4	5.0
$b_{3j}(in)$	$x_{15+7(i-1)+3}$	2.5	0.8	10.0
$b_{4j}(in)$	$x_{15+7(i-1)+4}$	3.0	0.4	10.0
$t_{2j}(in)$	$x_{15+7(i-1)+5}$	0.22	0.02	0.5
$t_{3j}(in)$	$x_{15+7(i-1)+6}$	0.22	0.01	0.25
$t_{4j}(in)$	$x_{15+7(i-1)+7}$	0.1	0.01	0.125

Table 26. Optimization Results for Wing Box using Exact Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, lb	761.26449	761.67282	
Objective function, lb	760.82450	761.26069	
Weight of truss, lb	11.25395	11.48045	
Weight of shear panels, lb	94.43436	94.22832	
Weight of OLSR panels, lb	655.13620	655.55200	
Number of ODMs	34	36	
Global analyses	1970	1647	16.40
Number of Evaluations			
Objective function	3193	3374	
Constraints	711	746	
Constraint gradients	34	36	
Approximate constraints	0	0	
CPU Times *			
Total	183.60	162.80	11.33
ODM	64.68	68.29	
Direction	118.59	94.06	
Hessian setup	3.55	3.59	
Solution	0.60	0.43	
Objective function	0.22	0.32	
Constraints	64.23	67.81	
Constraint gradients	113.55	82.11	27.69
Approximate constraints	0.00	0.00	

* CPU seconds on IBM 3084

Table 27. Optimization Results for Wing Box using Linear Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, lb	761.26253	761.63331	
Objective function, lb	760.82875	761.22815	
Weight of truss, lb	11.24802	11.37848	
Weight of shear panels, lb	94.41886	94.27374	
Weight of OLSR panels, lb	655.16180	655.57600	
Number of ODMs	35	38	
Global analyses	1505	1074	28.64
Number of Evaluations			
Objective function	3499	3685	
Constraints	209	123	
Constraint gradients	35	38	
Approximate constraints	735	788	
CPU Times *			
Total	141.88	112.38	20.79
ODM	19.51	11.89	
Direction	121.94	100.04	
Hessian setup	3.69	3.77	
Solution	0.59	0.43	
Objective function	0.27	0.35	
Constraints	18.84	11.24	
Constraint gradients	116.83	87.44	29.39
Approximate constraints	0.10	0.12	

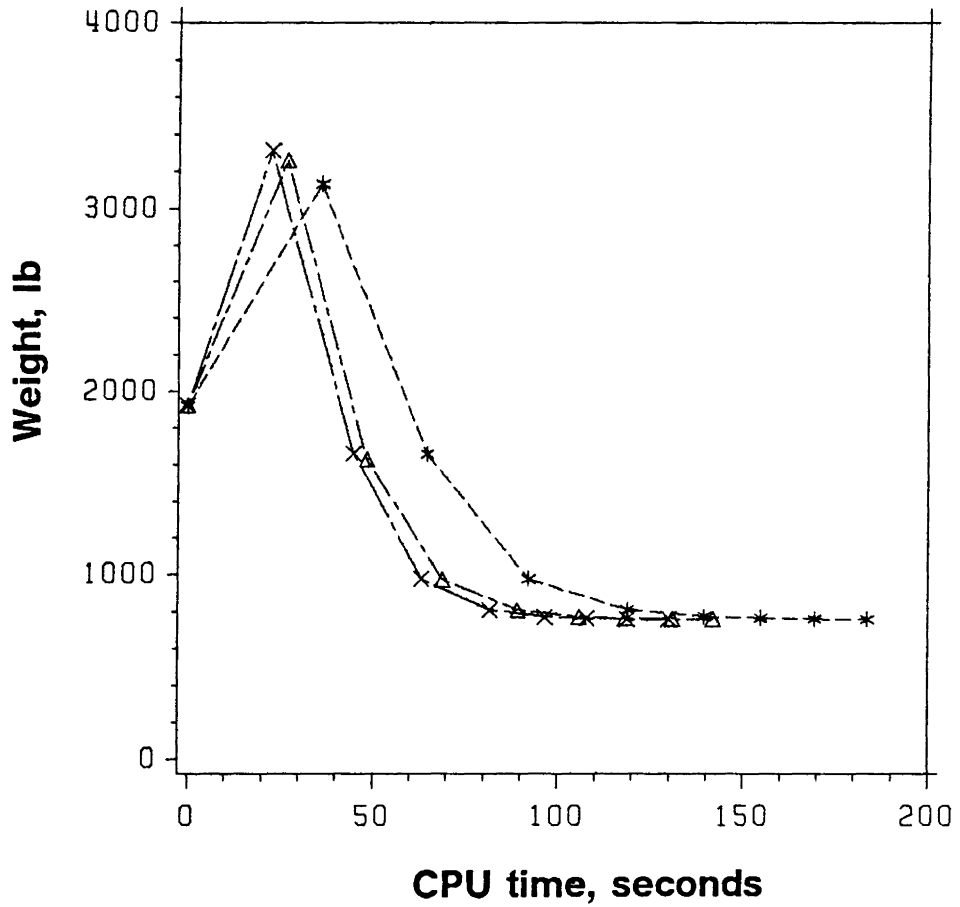
* CPU seconds on IBM 3084

Table 28. Optimization Results for Wing Box using Quadratic Approximation for Constraints

	Coupled System	Decoupled System	Percent Savings
Total function, lb	761.25809	761.66938	
Objective function, lb	760.81760	761.25891	
Weight of truss, lb	11.24686	11.48051	
Weight of shear panels, lb	94.43592	94.23850	
Weight of OLSR panels, lb	655.13480	655.53980	
Number of ODMs	34	38	
Global analyses	1367	1028	24.80
Number of Evaluations			
Objective function	3302	3575	
Constraints	71	77	
Constraint gradients	35	38	
Approximate constraints	676	724	
CPU Times *			
Total	129.88	106.60	17.92
ODM	6.99	7.59	
Direction	122.45	98.59	
Hessian setup	3.72	3.83	
Solution	0.62	0.47	
Objective function	0.30	0.31	
Constraints	6.48	6.95	
Constraint gradients	117.35	85.97	26.74
Approximate constraints	0.17	0.19	

* CPU seconds on IBM 3084

CONVERGENCE HISTORY FOR VARIOUS APPROXIMATIONS (WING BOX - 3 PANELS)



--* **Exact**
 △△△ **Linear**
 ××× **Quadratic**

Figure 27. Optimization Convergence History for Various Approximations for Wing Box

similar (Figure 27), the quadratic approximation required the least amount of CPU time. Since the initial design was infeasible there was an increase in the objective function during the first response surface, after which it steadily decreased until convergence.

Figure 28 shows an 11 percent reduction in total CPU time by using the decoupling technique, while reductions of 17-20 percent are obtained if constraint approximations are used (Tables 27 and 28). There is a 25-30 percent reduction in CPU time for constraint gradient evaluation (see Figure 29). Linear and quadratic approximations result in considerable savings compared to exact constraints. Figure 30 shows a 15 percent reduction in the number of global analyses for exact constraints, while the reduction is 30 percent for approximate constraints.

These savings are expected to increase as the number of subsystems (panels) are increased, since this increases the sparsity or decoupling between the subsystems. The addition of local constraints can also increase the savings in the evaluation of the hessian matrix.

CPU TIMES FOR COUPLED AND DECOUPLED SYSTEMS (WING BOX - 3 PANELS)

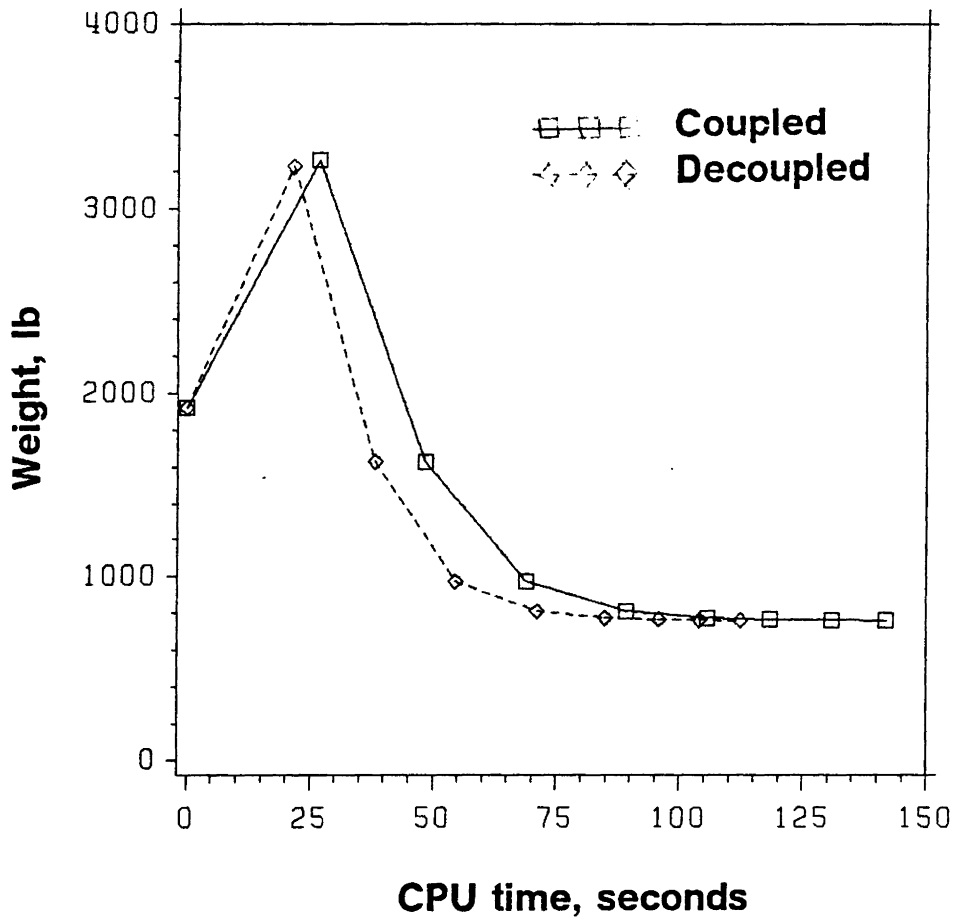


Figure 28. Optimization Convergence History for Coupled and Decoupled Systems with Exact Constraints for Wing Box

CPU TIMES FOR EXACT AND LINEAR APPROXIMATIONS (WING BOX - 3 PANELS)

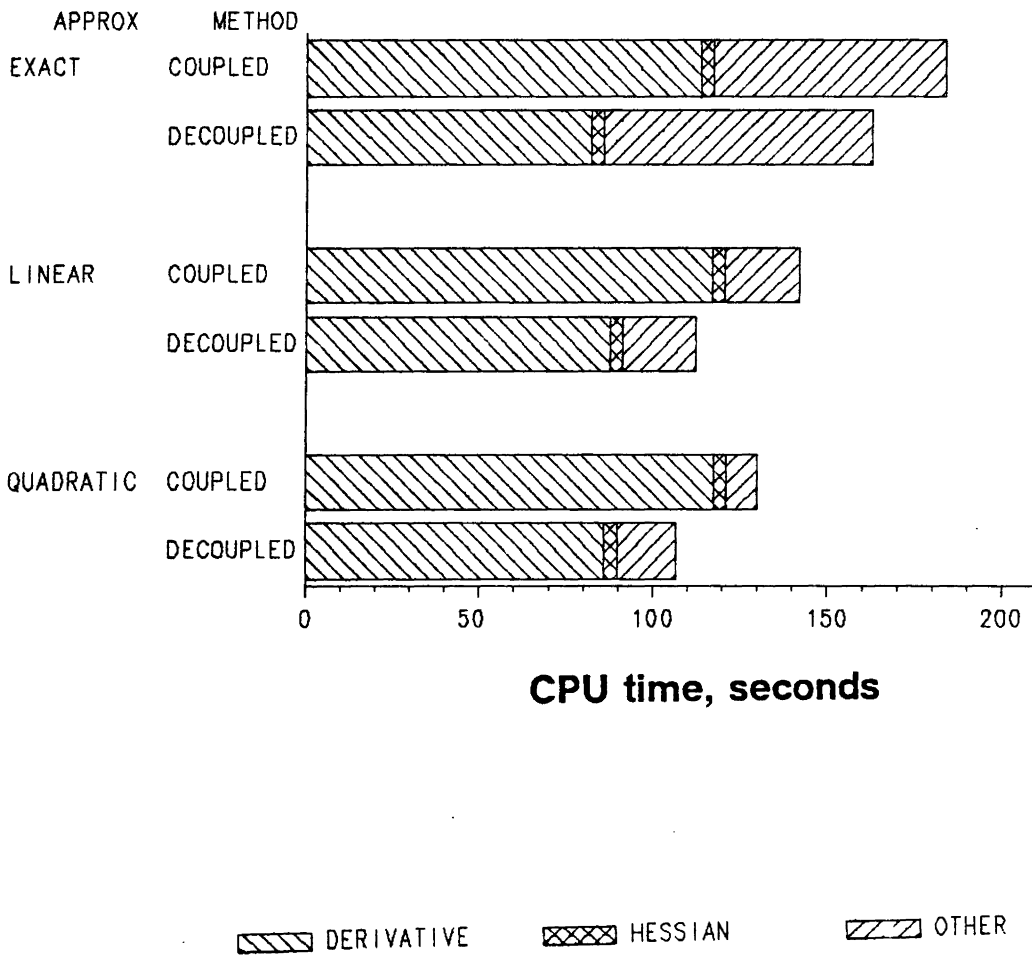


Figure 29. CPU Times for Coupled and Decoupled Systems with Various Approximations for Wing Box

NUMBER OF GLOBAL ANALYSES WITH VARIOUS APPROXIMATIONS (WING BOX - 3 PANELS)

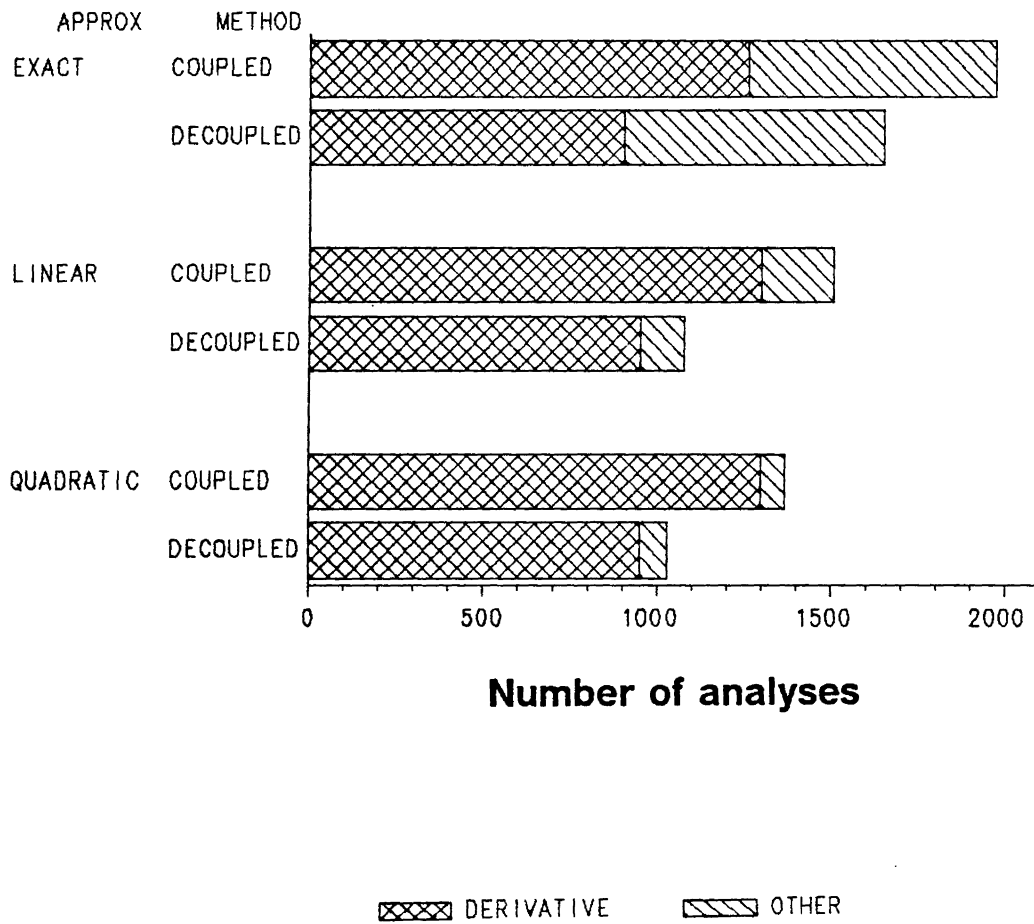


Figure 30. Number of Global Analyses for Coupled and Decoupled Systems with Various Approximations for Wing Box

8.0 Conclusions and Recommendations

Engineering design is hierarchical in nature, and if no attempt is made to benefit from this hierarchical nature, design optimization can be very expensive. There are two alternatives to taking advantage of the hierarchical nature of structural design problems. Multi-level optimization techniques incorporate the hierarchy at the formulation stage, and result in the coordinated optimization of a hierarchy of subsystems. The use of multi-level optimization techniques often necessitates the use of equality constraints. These constraints can sometimes cause numerical difficulties during optimization. Single-level decomposition techniques take advantage of the hierarchical nature to reduce the optimization cost.

In this research the decomposition approach has been followed to reduce the computational effort in a single-level design space. A decoupling technique has been developed that retains the advantages of a partitioned system of smaller independent subsystems without an increase in the total

number of design variables. A penalty function formulation using Newton's method for the solution of a sequence of unconstrained minimizations was employed. The optimization of the decoupled system is cheaper due to (i) cheaper evaluation of the hessian matrix by taking advantage of its sparsity, (ii) fewer global analyses for constraint derivative calculations, and (iii) utilizing the decoupled nature of the hessian matrix in the solution process. Further, the memory requirements of the decoupled system are much less than that of the original coupled system. These benefits increase substantially for design problems with larger and larger number of detailed design variables.

Orthotropic material properties as stiffness global variables have been shown to be effective as global variables for panels in a hierarchical wing design formulation.

The proposed decoupling technique was implemented to minimize the volume of a portal frame and a wing box. Computational savings of up to 50 percent have been obtained for medium sized problems. The savings increase as the size of the problem and the amount of decoupling is increased. The procedure is simple to implement. For truly large systems this decoupling technique provides the necessary reduction of computational effort to make the optimization process viable.

For further research more extensive applications of the decoupling method are recommended especially to large design problems. It is anticipated that

even greater savings can be achieved by the application of the proposed technique.

For the wing box example discussed in Chapter 7 additional local constraints can be added for the stresses in individual plies of the composite layers. Also, global and local buckling of the cover panels should be considered in order to make the design more realistic. The computer program PASCO (Ref 71) has these capabilities. An interface between the optimization program and PASCO can be developed to compute the stress and buckling constraints at the detailed design level.

Appendix A. Hessian Matrix in Decoupled Space

In this appendix some comments about the hessian matrix in the decoupled space are made. We shall consider a two subsystem configuration, though the development can be easily extended to an arbitrary number of subsystems. Also, the notation developed in Chapter 5 will be used extensively.

Let $f(R_1, E_1, R_2, E_2)$ be a general function in the coupled $E - R$ space, while $\bar{f}(R_1, D_1, R_2, D_2)$ is the same function in the transformed decoupled $E - R$ space. The global variable vectors are D_1 and D_2 , where the decoupling variable vectors D_1 and D_2 are used to eliminate the vectors E_1 and E_2 in the coupled system.

We therefore have

$$D_1 = D_1(R_1, E_1) \quad \text{and} \quad D_2 = D_2(R_2, E_2)$$

The hessian matrix corresponding to the second derivatives of f in the coupled space is

$$H = \begin{bmatrix} f_{R_1 R_1} & f_{R_1 E_1} & f_{R_1 R_2} & f_{R_1 E_2} \\ f_{R_1 E_1}^T & f_{E_1 E_1} & f_{E_1 R_2} & f_{E_1 E_2} \\ f_{R_1 R_1}^T & f_{E_1 R_2}^T & f_{R_2 R_2} & f_{R_2 E_2} \\ f_{R_1 E_1}^T & f_{E_1 E_2}^T & f_{R_2 E_2}^T & f_{E_2 E_2} \end{bmatrix} \quad [A.1]$$

The transformation matrix for this system is (see equations [5.13] and [5.20])

$$T = \begin{bmatrix} I & 0 & 0 & 0 \\ a_1 & b_1 & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & a_2 & b_2 \end{bmatrix} \quad [A.2]$$

where

$$\begin{aligned} a_1 &= -D_{1E_1}^{-1} D_{1E_1} \\ b_1 &= D_{1E_1}^{-1} \\ a_2 &= -D_{2E_2}^{-1} D_{2E_2} \\ b_2 &= D_{2E_2}^{-1} \end{aligned} \quad [A.3]$$

Carrying out the transformation of the hessian matrix from the coupled (H) to the decoupled (\hat{H}) system (see equation [5.17]), we have

$$\hat{H} = T^T H T \quad [A.4]$$

$$\hat{H} = \begin{bmatrix} \hat{f}_{R_1 R_1} & \hat{f}_{R_1 D_1} & \hat{f}_{R_1 R_2} & \hat{f}_{R_1 D_2} \\ \hat{f}_{R_1 D_1}^T & \hat{f}_{D_1 D_1} & \hat{f}_{D_1 R_2} & \hat{f}_{D_1 D_2} \\ \hat{f}_{R_1 R_2}^T & \hat{f}_{D_1 R_2}^T & \hat{f}_{R_2 R_2} & \hat{f}_{R_2 D_2} \\ \hat{f}_{R_1 D_2}^T & \hat{f}_{D_1 D_2}^T & \hat{f}_{R_2 D_2}^T & \hat{f}_{D_2 D_2} \end{bmatrix} \quad [A.5]$$

where

$$\hat{f}_{R_1 R_1} = f_{R_1 R_1} + \mathbf{a}_1^T f_{R_1 E_1}^T + f_{R_1 E_1} \mathbf{a}_1 + \mathbf{a}_1^T f_{E_1 E_1}^T \mathbf{a}_1$$

$$\hat{f}_{R_1 D_1} = f_{R_1 E_1} \mathbf{b}_1 + \mathbf{a}_1^T f_{E_1 E_1}^T \mathbf{b}_1$$

$$\hat{f}_{R_1 R_2} = f_{R_1 R_2} + \mathbf{a}_1^T f_{E_1 R_2}^T + f_{R_1 E_2} \mathbf{a}_2 + \mathbf{a}_1^T f_{E_1 E_2}^T \mathbf{a}_2$$

$$\hat{f}_{R_1 D_2} = f_{R_1 E_2} \mathbf{b}_2 + \mathbf{a}_1^T f_{E_1 E_2}^T \mathbf{b}_2$$

$$\hat{f}_{D_1 D_1} = \mathbf{b}_1^T f_{E_1 E_1}^T \mathbf{b}_1$$

$$\hat{f}_{D_1 R_2} = \mathbf{b}_1^T f_{E_1 R_2}^T + \mathbf{b}_1^T f_{E_1 E_2}^T \mathbf{a}_2$$

$$\hat{f}_{D_1 D_2} = \mathbf{b}_1^T f_{E_1 E_2}^T \mathbf{b}_2$$

$$\hat{f}_{R_2 R_2} = f_{R_2 R_2} + \mathbf{a}_2^T f_{R_2 E_2}^T + f_{R_2 E_2} \mathbf{a}_2 + \mathbf{a}_2^T f_{E_2 E_2}^T \mathbf{a}_2$$

$$\hat{\bar{f}}_{R_2 D_2} = f_{R_2 E_2} \mathbf{b}_2 + \mathbf{a}_2^T f_{E_2 E_2}^T \mathbf{b}_2$$

$$\hat{\bar{f}}_{D_2 D_2} = \mathbf{b}_2^T f_{E_2 E_2}^T \mathbf{b}_2$$

We shall now show that $\hat{\mathbf{H}}$ and \mathbf{H}^* have the same sparsity pattern.

For a single subsystem, recalling from Chapter 5 after linearizing the equality constraints and neglecting higher order terms, we have

$$\bar{f}_{DD} = \mathbf{b}^T f_{EE} \mathbf{b} \quad [\text{A.6}]$$

$$\bar{f}_{RD} = f_{RE} \mathbf{b} + \mathbf{a}^T f_{EE} \mathbf{b} \quad [\text{A.7}]$$

$$\bar{f}_{RR} = f_{RR} + f_{RE} \mathbf{a} + \mathbf{a}^T f_{RE}^T + \mathbf{a}^T f_{EE} \mathbf{a} \quad [\text{A.8}]$$

where

$$\mathbf{a} = -\mathbf{D}_E^{-1} \mathbf{D}_E$$

$$\mathbf{b} = \mathbf{D}_E^{-1}$$

The above expressions can be generalized for a system consisting of two subsystems. They are obtained from direct differentiation of the decoupled system, and hence correspond to the terms in the \mathbf{H}^* matrix.

Let us now show the similarity between the terms of $\hat{\mathbf{H}}$ and \mathbf{H}^* .

Note that

$$H^* = \begin{bmatrix} \bar{f}_{R_1R_1} & \bar{f}_{R_1D_1} & \bar{f}_{R_1R_2} & \bar{f}_{R_1D_2} \\ \bar{f}_{R_1D_1}^T & \bar{f}_{D_1D_1} & \bar{f}_{D_1R_2} & \bar{f}_{D_1D_2} \\ \bar{f}_{R_1R_2}^T & \bar{f}_{D_1R_2}^T & \bar{f}_{R_2R_2} & \bar{f}_{R_2D_2} \\ \bar{f}_{R_1D_2}^T & \bar{f}_{D_1D_2}^T & \bar{f}_{R_2D_2}^T & \bar{f}_{D_2D_2} \end{bmatrix} \quad [A.9]$$

From equation [A.8]

$$\begin{aligned} \bar{f}_{R_1R_1} &= f_{R_1R_1} + \mathbf{a}_1^T f_{R_1E_1}^T + f_{R_1E_1} \mathbf{a}_1 + \mathbf{a}_1^T f_{E_1E_1}^T \mathbf{a}_1 \\ &= \hat{f}_{R_1R_1} \end{aligned}$$

From equation [A.7]

$$\begin{aligned} \bar{f}_{R_1D_1} &= f_{R_1E_1} \mathbf{b}_1 + \mathbf{a}_1^T f_{E_1E_1}^T \mathbf{b}_1 \\ &= \hat{f}_{R_1D_1} \end{aligned}$$

From equation [A.8]

$$\begin{aligned} \bar{f}_{R_1R_2} &= f_{R_1R_2} + \mathbf{a}_1^T f_{E_1R_2}^T + f_{R_1E_2} \mathbf{a}_2 + \mathbf{a}_1^T f_{E_1E_2}^T \mathbf{a}_2 \\ &= \hat{f}_{R_1R_2} \end{aligned}$$

From equation [A.7]

$$\begin{aligned} \bar{f}_{R_1D_2} &= f_{R_1E_2} \mathbf{b}_2 + \mathbf{a}_1^T f_{E_1E_2}^T \mathbf{b}_2 \\ &= \hat{f}_{R_1D_2} \end{aligned}$$

From equation [A.6]

$$\begin{aligned}\bar{f}_{D_1 D_1} &= \mathbf{b}_1^T f_{E_1 E_1}^T \mathbf{b}_1 \\ &= \hat{f}_{D_1 D_1}\end{aligned}$$

From the transpose of equation [A.7]

$$\begin{aligned}\bar{f}_{D_1 R_2} &= \mathbf{b}_1^T f_{E_1 R_2}^T + \mathbf{b}_1^T f_{E_1 E_2}^T \mathbf{a}_2 \\ &= \hat{f}_{D_1 R_2}\end{aligned}$$

From equation [A.6]

$$\begin{aligned}\bar{f}_{D_1 D_2} &= \mathbf{b}_1^T f_{E_1 E_2}^T \mathbf{b}_2 \\ &= \hat{f}_{D_1 D_2}\end{aligned}$$

From equation [A.8]

$$\begin{aligned}\bar{f}_{R_2 R_2} &= f_{R_2 R_2} + \mathbf{a}_2^T f_{R_2 E_2}^T + f_{R_2 E_2} \mathbf{a}_2 + \mathbf{a}_2^T f_{E_2 E_2}^T \mathbf{a}_2 \\ &= \hat{f}_{R_2 R_2}\end{aligned}$$

From equation [A.7]

$$\begin{aligned}\bar{f}_{R_2 D_2} &= f_{R_2 E_2} \mathbf{b}_2 + \mathbf{a}_2^T f_{E_2 E_2}^T \mathbf{b}_2 \\ &= \hat{f}_{R_2 D_2}\end{aligned}$$

From equation [A.6]

$$\begin{aligned}\bar{f}_{D_2 D_2} &= \mathbf{b}_2^T \hat{f}_{E_2 E_2}^T \mathbf{b}_2 \\ &= \hat{f}_{D_2 D_2}\end{aligned}$$

Further note that by definition, as a result of the introduction of the decoupling variable vectors \mathbf{D}_1 and \mathbf{D}_2

$$\bar{f}_{R_1 R_2} = 0 = \hat{f}_{R_1 R_2}$$

Hence \mathbf{H}^* and $\hat{\mathbf{H}}$ have the same sparsity pattern. Though the above expressions are approximations because the higher order terms in the linearized equality constraints are neglected, the following proof for sparsity can be shown to be valid for the exact transformations from the coupled to the decoupled system.

References

1. Giles, G.L.: "Procedure for Automated Aircraft Wing Structural Design," **Journal of the Structural Division, ASCE**, Vol 97, 1971, pp 99-113
2. Sobieszczanski, J. and Loendorf, D.D.: "A Mixed Optimization Method for Automated Design of Fuselage Structures," **Journal of Aircraft**, Vol 9, 1972, pp 805-811
3. Loendorf, D.D., Sobieszczanski, J. and Stroud, W.J.: "Computer-Aided Design Study of All-Metal and Composite Stiffened Metal Fuselage Structures," AIAA Paper No 73-373, AIAA/ASME/SAE 14th Structures, Structural Dynamics and Materials Conference, Williamsburg, Virginia, April 1973
4. Arora, J.S. and Govil, A.K.: "An Efficient Method for Optimal Structural Design by Substructuring," **Computers and Structures**, Vol 7, 1977, pp 507-515
5. Sobieszczanski-Sobieski, J.: "An Integrated Computer Procedure for Sizing Composite Airframe Structures," NASA TP-1300, 1979
6. Kirsch, U., Reiss, M. and Shamir, U.: "Optimum Design by Partitioning into Substructures," **Journal of the Structural Division, ASCE**, 1972, pp 249-261
7. Noor, A.K., Kamel, H.A. and Fulton, R.E.: "Substructuring Techniques — Status and Projections," **Computers and Structures**, Vol 8, No 5, May 1978, pp 621-632

8. Hughes, O.F., Mistree, F. and Zanic, V.: "A Practical Method for the Rational Design of Ship Structures," **Journal of Ship Research**, Vol 24, 1980, pp 101-113
9. Dantzig, G.B.: **Linear Programming and Extensions**, Princeton University Press, Princeton, New Jersey, 1963
10. Rosen, J.B.: "Primal Partition Programming for Block Diagonal Matrices," **Numerische Mathematik**, Vol 6, 1964, pp 250-260
11. Lasdon, L.S.: **Optimization Theory for Large Systems**, MacMillan Series in Operations Research, MacMillan Publishing Co., Inc., 1970
12. Mesarovic, M.D., Macko, D. and Takahara, Y.: **Theory of Hierarchical Multi-Level Systems**, Academic, New York, 1970
13. Kirsch, U.: **Optimum Structural Design**, McGraw-Hill Book Company, 1981, pp 198-215
14. Kirsch, U.: "Multilevel Approach to Optimum Structural Design," **Journal of the Structural Division**, ASCE, 1975, pp 957-974
15. Davis, W.J.: "Generalized Decomposition Procedure and its Application to Engineering Design," ASME Publication, Paper No 77-DET-62, 1977
16. Ginsburg, S. and Kirsch, U.: "Multilevel Optimization of Arrays of Protective Structures," International Symposium on Optimum Structural Design, Tucson, Arizona, 1981, pp 5-33 - 5-39
17. Haftka, R.T.: "An Improved Computational Approach for Multilevel Optimum Design," **Journal of Structural Mechanics**, Vol 12(2), 1984, pp 245-261
18. Ho, J.K.: "Optimal Design of Multi-stage Structures: A Nested Decomposition Approach," Systems Optimization Laboratory, Department of Operations Research, Stanford University, California, Technical Report 74-2, 1974
19. Kalliauer, A.: "An Algorithm for Hierarchical Optimization of Large-Scale Problems with Nested Structures," **Mathematical Programming**, Vol 25, 1983, pp 25-45
20. Kaneko, I. and Ha, C.D.: "A Decomposition Procedure for Large-Scale Optimum Plastic Design Problems," **International Journal for Numerical Methods in Engineering**, Vol 19, 1983, pp 873-889

21. Kirsch, U.: "Multilevel Optimal Design of Reinforced Concrete Structures," **Engineering Optimization**, 1983, Vol 6, pp 207-212
22. Kirsch, U. and Moses, F.: "Decomposition in Optimum Structural Design," **Journal of the Structural Division**, Proceedings of the American Society of Civil Engineers, ST1, 1979, pp 85-100
23. Kirsch, U.: "Synthesis of Structural Geometry Using Approximation Concepts," **Computers and Structures**, Vol 15, No 3, 1982, pp 305-314
24. Kirsch, U.: "Synthesis of Elastic Structures with Controlled Forces," **Computers and Structures**, Vol 6, 1976, pp 111-116
25. Lev, O.E.: "On the Solution of a Nonlinear Programming Problem by Decomposition," **Journal of Optimization Theory and Applications**, Vol 22, No 1, 1977, pp 31-34
26. Lust, R.V. and Schmit, L.A.: "Alternative Approximation Concepts for Space Frame Synthesis," AIAA Paper No 85-0696, AIAA/ASME/ASCE/AHS 26th Structures, Structural Dynamics and Materials Conference, Orlando, Florida, April 1985
27. Mahmoud, M.S.: "Multilevel Systems Control and Applications: A Survey," **IEEE Transactions on Systems, Man, and Cybernetics**, Vol SMC-7, No 3, March 1977, pp 125-143
28. Schmit, L.A. and Ramanathan, R.K.: "Multilevel Approach to Minimum Weight Design Including Buckling Constraints," **AIAA Journal**, Vol 16, No 2, 1978, pp 97-104
29. Schmit, L.A. and Chang, K.J.: "A Multilevel Method for Structural Synthesis," AIAA Paper No 84-0853, AIAA/ASME/ASCE/AHS 24th Structures, Structural Dynamics and Materials Conference, Palm Springs, California, May 1984
30. Schmit, L.A. and Mehrinfar, M.: "Multilevel Optimum Design of Structures with Fiber-Composite Stiffened-Panel Components," **AIAA Journal**, Vol 20, No 1, 1982, pp 138-147
31. Mehrinfar, M.: "Multilevel Optimum Design of Wing Box Structures with Fiber Composite Panel Components," Ph. D. Dissertation, University of California, Los Angeles, California, 1980
32. Sobieszczanski-Sobieski, J.: "A Linear Decomposition Method for Large Optimization Problems - Blueprint for Development," NASA Technical Memorandum TM-83248, Feb 1982

33. Sobieszczanski-Sobieski, J., James, B. and Dovi, A.: "Structural Optimization by Multilevel Decomposition," AIAA Paper No 83-0832, AIAA/ASME/ASCE/AHS 24th Structures, Structural Dynamics and Materials Conference, Lake Tahoe, Nevada, May 1983
34. Sobieszczanski-Sobieski, J., Barthelemy, J.F.M. and Giles, G.L.: "Aerospace Engineering Design by Systematic Decomposition and Multilevel Optimization," NASA TM 85823, June 1984
35. Sobieszczanski-Sobieski, J., James, B. and Riley, M.F.: "Structural Optimization by Generalized Multilevel Optimization," AIAA Paper No 85-0697, AIAA/ASME/ASCE/AHS 26th Structures, Structural Dynamics and Materials Conference, Orlando, Florida, April 1985
36. Tatjewski, P.: "Dual Methods of Multilevel Optimization," *Academie Polonaise des Sciences, Bulletin, Serie des Sciences Techniques*, Vol 25, No 3, 1977, pp 9(247)-16(254), also A77-46588, *International Aerospace Abstracts*, AIAA, 1977
37. Twisdale, L.A. and Khachaturian, N.: "Multistage Optimization of Structures," **Journal of the Structural Division**, ST5, 1975, pp 1005-1020
38. Vanderplaats, G.N.: "Structural Optimization Via a Design Space Hierarchy," **International Journal for Numerical Methods in Engineering**, Vol 10, No 3, 1976, pp 713-717
39. Woo, T.H.: "Decomposition in Structural Synthesis," Ph. D. Dissertation, University of California, Los Angeles, California, 1977
40. Hughes, O.: "A Method for Nonlinear Optimum Design of Large Structures and Applications to Naval Ship Design," **Proceedings of the International Symposium on Optimum Structural Design**, Tuscon, Arizona, Oct 1981
41. Sobieszczanski-Sobieski, J., Barthelemy, J.F. and Riley, K.M.: "Sensitivity of Optimum Solutions to Problem Parameters," AIAA Paper No 81-0548R, also **AIAA Journal**, Vol 20, No 9, September 1982, pp 1291-1299
42. Barthelemy, J.F.: "Development of a Multilevel Optimization Approach to the Design of Modern Engineering Systems," Ph. D. Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, June 1983, also NASA CR-172184, August 1983

43. Barthelemy, J.F. and Sobieszczanski-Sobieski, J.: "Extrapolation of Optimum Design Based on Sensitivity Derivatives," **AIAA Journal**, Vol 21, 1983, pp 797-799
44. Schmit, L.A. and Chang, K.J.: "Optimum Design Sensitivity Based on Approximation Concepts and Dual Methods," AIAA Paper No 82-0713-CP, Presented at the AIAA/ASME/ASCE/AHS 23rd Structures, Structural Dynamics and Materials Conference, New Orleans, Louisiana, May 1982 **AIAA Journal**, Vol 21, 1983, pp 797-799
45. Barthelemy, J.F. and Sobieszczanski-Sobieski, J.: "On Optimum Sensitivity Derivatives of Objective Functions in Nonlinear Programming," **AIAA Journal**, Vol 21, 1983, pp 913-915
46. Haug, E.J., Komkov, V.J. and Choi, K.K.: "Design Sensitivity Analysis of Structural Systems," Academic, New York (in press)
47. Haftka, R.T.: "Damage Tolerant Design Using Collapse Techniques," AIAA Paper 82-0718-CP, Presented at the AIAA/ASME/ASCE/AHS 23rd Structures, Structural Dynamics and Materials Conference, New Orleans, Louisiana, May 1982
48. Miura, H. and Schmit, L.A.: "NEWSUMT - A Fortran Program for Inequality Constrained Function Minimization - User's Guide," NASA CR-159070, June 1979.
49. Haftka, R.T. and Starnes, J.H.: "Applications of a Quadratic Extended Interior Penalty Function for Structural Optimization," **AIAA Journal**, Vol 14, June 1976, pp 718-724
50. Vanderplaats, G.N.: "ADS - A Fortran Program for Automated Design Synthesis : Version 1.00," NASA Contractor Report 172460, October 1984
51. Vanderplaats, G.N.: "CONMIN - A Fortran Program for Constrained Minimization - User's Manual," TM X-62282, NASA, August 1973.
52. Gabriele, G.A., and Ragsdell, K.M.: "OPT A Nonlinear Programming Code in Fortran IV - Users Manual," Purdue Research Foundation, West Lafayette, Indiana, 1976.
53. Dwyer, W.J., Emerton, R.K., and Ojalvo, I.U.: "An Automated Procedure for the Optimization of Practical Aerospace Structures, Vol I - Theoretical Development and User's Information," AFFDL-TR-70118, U.S.Air Force, April 1971.

54. Ashley, H.: "On Making Things The Best - Aeronautical Uses of Optimization," **Journal of Aircraft**, Vol 19, No 1, 1982, pp.5-28.
55. Schmit, L.A., Jr., and Farshi, B.: "Some Approximation Concepts for Structural Synthesis," **AIAA Journal**, Vol 12, May 1974, pp. 692-699.
56. Noor, A.K.: "Multiple Configuration Analysis via Mixed Method," **J. of the Structural Div., ASCE**, Vol 100, No ST 9, 1974, pp. 1991-1997.
57. Noor, A.K., and Lowder, H.E.: "Structural Reanalysis Via a Mixed Method," **Computers and Structures**, Vol 5, April 1975, pp. 9-12.
58. Austin, Fred.: "A Rapid Optimization Procedure for Structures Subjected to Multiple Constraints," AIAA/ASME 18th Structures, Structural Dynamics and Materials Conference, March 1977, pp. 71-79. (Available as AIAA Paper No. 77-374).
59. Storaasli, Olaf O., and Sobieszczanski, Jaroslaw.: "On the Accuracy of the Taylor Approximation for Structure Resizing," **AIAA J.**, Vol 12, No 2, February 1974, pp. 231-233.
60. Starnes, J.H., Jr., and Haftka, R.T.: "Preliminary Design of Composite Wings for Buckling, Strength and Displacement Constraints," **Journal of Aircraft**, Vol 16, No 8, August 1979, pp. 564-570.
61. Haftka, R.T., and Shore, C.P.: "Approximation Methods for Combined Thermal/Structural Design," TP-1428, NASA, June 1979.
62. Miura, H., and Schmit, L.A.: "Second Order Approximation of Natural Frequency Constraints in Structural Synthesis," **Int. J. Num. Meth. Eng.**, Vol 13, 1978, pp. 337-351.
63. Proceedings, International Symposium on Optimum Structural Design (Atrek, A. and Gallagher, R.H., eds.) University of Arizona, Tucson, Arizona, October 1981.
64. Kirsch, U., and Hofman, B.: "Approximate Behavior Models for Optimum Structural Design," International Symposium on Optimum Structural Design, *ibid*, p. 7-17.
65. Bennett, J.A.: "Application of Linear Constraint Approximations to Frame Structures," International Symposium on Optimum Structural Design, *ibid*, p. 7-9.
66. Halmos, E.: "About Determination Optimal-Weight Statically Indeterminate Bar Structures of Complex Stress by Using Explicit

Forms of the Conditions," International Symposium on Optimum Structural Design, *ibid*, p. 7-37.

67. Esping, B.: "Weight Optimization of Membrane Structures," International Symposium on Optimum Structural Design, *ibid*, p. 7-27.
68. Wang, B.P., and Pilkey, W.D.: "Parametrization in Finite Element Analysis," International Symposium on Optimum Structural Design, *ibid*, p. 7-1.
69. Queau, J.P., and Trompette, P.: "Optimal Shape Design of Turbine and Compressor Blades," International Symposium on Optimum Structural Design, *ibid*, p. 5-9.
70. Middleton, J.: "Optimal Shape Design to Minimize Stress Concentration Factors in Pressure Vessel Components," International Symposium on Optimum Structural Design, *ibid*, p. 5-23.
71. Stroud, W.J., and Anderson, M.S.: "PASCO: Structural Panel Analysis and Sizing Code, Capability and Analytical Foundations," TM-80181, NASA, November 1981.
72. Fiacco, A.V., and McCormick, G.P.: **Nonlinear Programming, Sequential Unconstrained Minimization Techniques**, John Wiley and Sons, Inc., New York, 1968.
73. Haftka, R.T.: "Automated Procedure for Design of Wing Structures to Satisfy Strength and Flutter Requirements," TN D-7264, NASA, July 1973.
74. Chu., C.S., et.al: "Finite Element Program to Analyze Cracked Orthotropic Sheets," CR-2698, NASA, July 1976.
75. Schmit, L.A., Jr., and Miura., H.: "Approximation Concepts for Efficient Structural Synthesis," CR-2552, NASA, March 1976.
76. Thareja, R.R. and Haftka, R.T.: "NEWSUMT-A - A Modified Version of NEWSUMT for Inequality and Equality Constraints - Version 1," VPI Report 148, Aerospace and Ocean Engineering Department, Virginia Tech, June 1985.
77. Grandhi, R.V., Thareja, R.R., and Haftka, R.T.: "NEWSUMT-A: A General Purpose Program for Constrained Optimization using Constraint Approximations," 1983 ASME International Computers in Engineering Conference and Exhibit, Chicago, August 1983, also 10th Design Automation Conference, Cambridge, Massachusetts, October

1984, **Journal of Mechanisms, Transmissions, and Automation in Design**, Vol 107 (1), Special Issue, March 1985, pp. 94 - 99.

78. Thareja, R.R. and Haftka, R.T.: "Numerical Difficulties Associated With Using Equality Constraints to Achieve Multi-Level Decomposition in Structural Optimization," AIAA Paper No. 86-0854-CP, AIAA/ASME/ASCE/AHS 27th Structures, Structural Dynamics and Materials Conference, San Antonio, Texas, May 1986.
79. Haftka, R.T.: "WIDOWAC Program Manual" NASA TM X-3071, October 1974.

**The vita has been removed from
the scanned document**