Zeitschrift: IABSE congress report = Rapport du congrès AIPC = IVBH Kongressbericht

Band: 10 (1976)

Rubrik: Theme II: Progress in structural optimization

II

Progrès dans l'optimisation structurale

Fortschritte in der Optimierung von Tragwerken

Progress in Structural Optimization

Leere Seite
Blank page
Page vide

Optimization Concepts and Techniques in Structural Design

Concepts et techniques d'optimisation

Grundlagen und Methoden

ANDREW B. TEMPLEMAN
Professor of Civil Engineering
University of Liverpool
Liverpool, GB

1. INTRODUCTION

Since the early 1960's structural optimization has established itself as an important area of research in structural engineering and it is now gradually becoming an integral part of the technological expertise of practising designers. The phrase "structural optimization" conjures a very specific impression in the minds of most engineers relating to the application of numerical search techniques to mathematical models of certain types of structures. Whilst this impression is correct in detail it is by no means the whole story. The history of architecture and structural design has been characterized from the very earliest times by efforts to design structures which were in some respect "better" than their predecessors; to use materials and resources more efficiently and to design structures which were as well-suited as possible to their functions. This continual search for improvement is essentially an optimization process, to use the term in its widest sense, and it pervades all present-day engineering activities.

On a national and regional scale governments seek to invest in new industries, resources and communications so as to obtain maximum benefit from a limited investment. At the project level an engineer strives to produce efficient, cheap and reliable designs for projects to satisfy functional requirements. The contractor wishes to use his resources of manpower, machinery and capital in the most efficient manner and the client is concerned whether his structure will be economical to run, to use and to maintain. The whole process of project planning, design, construction and operation is governed by the need to produce the best possible solution. Traditionally, emphasis has been placed on the intuitive skills and ability of experienced planners and engineers to produce near-optimal solutions to problems. However, the demands of an increasingly technologically-based population now require that projects are larger, more expensive and more complex. Financial pressures towards cost economy are increasing. Restrictions imposed by aesthetic, social, environmental and technological factors are ever more stringent. A result of this increasing complexity of the design process is that traditional reliance upon the skills of individual designers must change to meet present-day circumstances.

Consequently, more assistance is now sought from the digital computer, with its speed and power to solve complex problems rapidly, to meet these demands.

Since the digital computer was first used on structural engineering problems over twenty years ago it has completely transformed methods of structural analysis, so that now the use of matrix stiffness and flexibility methods as well as finite element techniques is commonplace in structural design offices throughout the world. However, as it became possible to analyse increasingly complex structures it also became more difficult to interpret the results of the analysis in a logical way for design purposes. It was, therefore, a natural trend in research to try to use the computer to produce design-oriented information rather than analysis-oriented information for the designer to interpret. Thus over the last ten or fifteen years research into aspects of computer-aided structural design has considerably increased. Once it was recognized that the engineering planning and design process is an optimum-seeking one it was also natural to see whether the computer could not also be used to produce not merely design information but optimum design information so that the structural engineer could play his part effectively in the planning and design of efficient, cheap and reliable structures. The computer, therefore, afforded the possibility of helping with the two basic problems mentioned above of the increasing technological complexity of structural design and the growing pressures towards cost control and economical designs.

However, just as the computer revolutionized structural analysis by demanding new techniques suited to its abilities a similar revolution was needed in design techniques if the computer was to be of real use in this area. In the early 1960's Schmit[1] and others gave a first insight into a new computer-oriented approach to structural design by examining the application of newly-developed techniques of mathematical programming and optimization to problems of structural design. Thus structural optimization in its modern form was born. With the passing of time it is perhaps difficult to realize that in the early 1960's the philosophy of structural optimization was really very novel. The early research referred to above clearly demonstrated what had not previously been self-evident: That the largely ad hoc processes of structural design which appeared to have little formal logic may in fact be expressed formally and in mathematical terms, and that there was as much rigour and logic in solving an optimum structural design problem as there was in solving a structural analysis problem. Perhaps the largest contribution which the study of optimization has given to structural engineering is that it has put structural design on a formal, mathematical basis and by so doing has unified a previously fragmentary and ill-disciplined subject.

Coming some fifteen years after Schmit's original work this paper examines some of the more recent developments in structural optimization techniques. It shows that there are still many difficulties to overcome for the full potential of optimization to be realized and it is hoped that it will give an insight into the way in which structural optimization creates a greater understanding of the nature of the design process and of structural behaviour.

2. A HIERARCHY OF STRUCTURAL OPTIMIZATION PROBLEMS

The rationalizing and formalizing influence which optimization has had upon structural design now means that structural optimization problems may be conventionally expressed as:

$$\left.\begin{array}{llll} \text{Minimize, or Maximize} & g_o\,(x_i) & & i = 1, \ldots, N \\ \text{Subject to constraints} & g_j\,(x_i) \begin{Bmatrix} < \\ = \\ > \end{Bmatrix} 0 & & j = 1, \ldots, M \\ & x_i \geq 0 & & i = 1, \ldots, N \end{array}\right\} \quad (1)$$

The variables in this problem, x_i, $i = 1, \ldots, N$, usually represent physical parameters of the structure to be designed such as dimensions, spacings, bar sizes, plate thicknesses, etc. These variables are under the designer's control and he wishes to find "best", or optimal values for all of them. The objective function $g_o\,(x_i)$ which is to be extremized represents some evaluable criterion of efficiency of the structure. The efficiency criteria most frequently used are such things as minimum structural weight, minimum cost, maximum factor of safety, etc. The choice of objective function will be commented upon later. The M constraints $g_j\,(x_i)$ may be equalities or inequalities and they originate from many sources. They specify, for example, the mechanical behaviour of the structure under load, the known properties of the materials used, requirements of relevant codes of practice, fabricational requirements, geometric and layout requirements, etc. All relevant restrictions and requirements upon the structure must appear among the constraints. The non-negativity condition upon each variable is necessary to ensure that values are obtained for all the problem variables which are real and feasible in an engineering sense. Sometimes integer values or values from a discrete set may be required.

Generally structural optimization problems are large and non-linear except for all but the simplest structures. This is seen if a single, very much simplified structural element is examined. Fig. 1 shows the cross-sectional shape of a typical reinforced concrete T-beam.

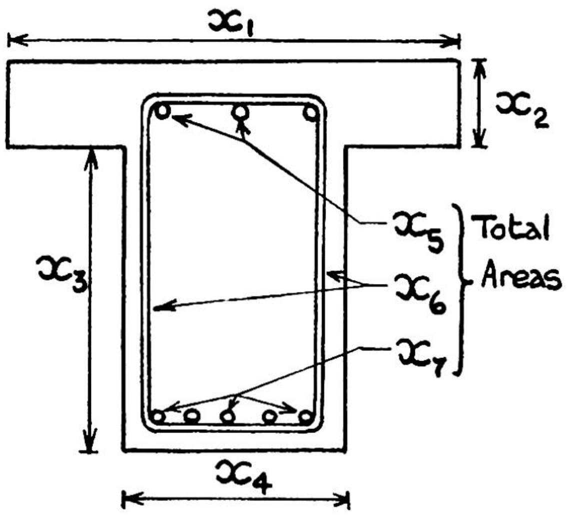


FIG. 1

To design this beam of known length and loading so as to minimize a simple cost function involves finding optimal values for the seven variables shown. The cost function g_o is fairly simple to write down. It involves the cost of the concrete and of the steel, both roughly proportional to the volumes of concrete and steel, and it also involves the cost of shuttering the beam during pouring. This cost depends upon the perimeter of the cross-section. The objective function is then a simple one but is non-linear in the variables. Many constraints are necessary to ensure that the concrete and steel can adequately carry the bending stresses in the beam and also resist shearing stresses. Codes of practice prescribe maximum permissible values for these stresses and also prescribe a large number of other permissible values for such things as deflections, crack widths, bar spacings, etc. There may be fabricational constraints which limit, for example, the bar areas to be not less than some available size. The designer may also add constraints himself if the beam has to fit some restricted location within a structure. Typically to fully describe the optimum design problem, between ten and twenty constraints may be needed in the seven variables and because of the functions involved in these constraints they are almost all highly non-linear ones.

Problems of this size and complexity can now be solved fairly rapidly by a variety of methods which will be outlined later in this paper. However, this example is merely a single beam element. Most real-world structures have many elements such as beams, columns, slabs, panels, etc., and to fully describe the optimum design problem for such real-world structures may require many hundreds of variables and constraints. The general characteristics of structural optimization problems are therefore that they are large, multivariate, non-linear, constrained problems.

Because problem size and complexity become enormous when all variables and constraints are lumped together in a single problem a hierarchy of problems has developed. The hierarchy is as follows:

1. Topology of the structure
2. Geometry of the structure
3. Overall sizes of structure members
4. Detailed design of elements

The logic and implications of this hierarchy can be demonstrated by reference to the beam example which falls into category 4 of the hierarchy as it is a detailed design of an element. Suppose this beam was one of a known number of beams supporting a deck slab longitudinally. By optimizing the cost of each beam in turn cost savings would accrue over a non-optimized design and may be considerable. However, the inquiring designer will ask himself whether the specified number of beams is itself optimal. If more longitudinal beams were used the load to be carried by each would be reduced and so the necessary size would also be reduced. The cost savings to be gained by using a larger number of smaller beams or a smaller number of larger beams might well be far greater than anything achieved by merely paring down upon detailed design sizes. This type of problem falls into category 3 of the hierarchy.

If the deck and beams form part of a bridge over several piers then the interested designer will soon find himself considering the cost savings to be made by varying the distances between piers to arrive at an optimum geometric arrangement in category 2 of the hierarchy. This promises the possibility of even greater cost economy. Finally category 1 which holds out the greatest savings of all, is concerned with topology. Was the decision to build a deck on concrete beams over supporting piers itself optimal? Would not an alternative structural form be more efficient? Why not use a steel box-section deck?

This example demonstrates the logic of the hierarchy fairly well and also demonstrates two general features of it. Starting from the lowest category 4, the higher up the hierarchy that optimization can be used the greater the potential for economy becomes but also the more difficult optimization becomes to implement. This second feature is reflected by the fact that very little work has been published on the optimum topologies of real-world structures, the work of Michell[2] being of theoretical interest rather than practical use. Indeed, the present state of the art of structural optimization is that a vast amount of research work has been published on methods for category 4 problems, and it can be stated that most problems in this category can be solved fairly rapidly. Much work has also been done in category 3 but success there has been less general and new techniques are required. The available literature in category 2 is fairly small but has significantly increased in the last two or three years. Much more work remains to be done in this category and in category 1 where significant practical literature is almost non-existent.

3. APPROXIMATION CONCEPTS

Very often approximations are used in the formulation of optimum design problems for two reasons. Firstly they can be used to hold problem sizes down to a level at which computer solution does not become inordinately expensive. Secondly they can be used to decompose a large design problem into a series of smaller problems. Another simple example demonstrates this. Consider the problem of the optimum elastic design of the beam/column framework of a multi-storey building such as that in Fig. 2. Each member of this framework may have up to ten detailed dimension variables and perhaps ten to twenty constraints. A complete optimum design problem for such a structure including both overall member sizes and detailed dimensions (i.e., categories 3 and 4) would therefore involve hundreds of variables and constraints and would be impossibly expensive in computation time.

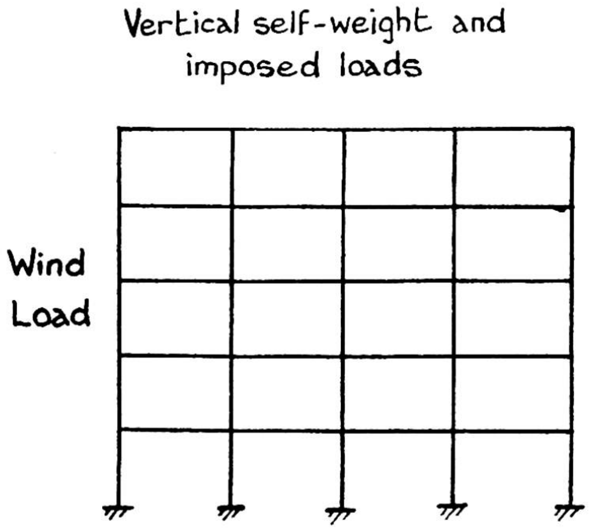


FIG. 2

Approximations can be used to effectively decompose or separate out the large problem into a single category 3 program and many simple category 4 problems. This is done by selecting for each beam or column element of the structure a single parameter and relating the cost, weight and behaviour of the element to this single parameter. In this particular example a suitable parameter might be stiffness, E I. Each element has thus been approximated and obviously in problems generally care must be taken to ensure that the approximate element really does behave as the real element does. The framework is then reassembled using approximate elements and a single category 3 problem may be formulated. This would consist of finding a complete set of optimal stiffness values for the framework which minimizes the cost of the framework while satisfying as constraints the equations of equilibrium and compatibility for all the applied load cases.

Having found the set of optimal stiffness values an analysis of the frame using these values determines all the beam and column moments and forces. Each element may then be designed separately as a category 4 problem so that its cost is minimized over all five to ten detailed variables. The loadings are those obtained in the analysis and the constraints would be the normal detailed constraints plus an additional one that the stiffness of the element must be equal to the value obtained for the approximate stiffness parameter in the overall size optimization.

Decomposition by means of approximations such as this is very widely used with considerable success. Frequently, in the aerospace industry very complex structures such as wing boxes, fin and tail structures, etc., are approximated as assemblies of membrane plates and shear panels for which rapid minimum weight design programs are available. One note of warning however; the last stage of any optimum design problem should always be a detailed analysis of the optimum design to ensure that approximations have not caused errors to be introduced. Indeed, on a more general basis it should perhaps be made clear that optimization is not intended to replace the designer. The objective is to provide him with information on what the most efficient solution to the problem posed might be. Only the designer can know whether the problem he posed is a complete one and whether it was representative of the real-world structure. Optimization can only suggest a possible design which the designer is then at liberty to modify or reject or accept after further analysis. There are always factors in the mind of a designer which he cannot quantify but which nevertheless influence his designs. In using structural optimization to assist him the designer should attempt to formulate a representative problem only. According to how much he feels has been put into the problem or omitted he alone will know how much reliance to place upon the suggested optimum design and also how much he must alter it to satisfy himself regarding the unquantifiable factors such as aesthetics. Structural optimization produces preliminary design information and should never be expected to produce a final design.

In this section on approximation concepts it is useful to comment further upon the choice of objective function to be extremized. In purely technical problems involving reliability, dynamic response, etc., the correct objective function is usually fairly obvious. In more general problems cost or weight usually provide a direct measure of the efficiency of a design. In the aerospace and shipbuilding fields weight is generally of paramount importance partly because it directly reflects an element of cost but more because it directly affects the operational efficiency of the vehicle. In civil engineering weight is perhaps not so important and cost seems to be the vital factor in design. Some structural design applications of optimization in the civil field are frequently criticized because weight has been used as an objective function or, if cost has been used then not enough of the elements of cost arising from fabrication, erection, labour, etc., have been used or the cost coefficients are invalid in an inflating economy. Sometimes these criticisms are valid but frequently the solution is relatively insensitive to variations in the objective function. What is important in constructing an objective function is to ensure that all the variable major elements are included with coefficients of the right order. If, in the future the price of steel was to reach the present price of gold then designs for steel structures would change only minimally since labour costs would likewise have escalated along with costs of all other materials. Cost objective functions are almost always approximations in themselves and as was stated earlier the important thing is to ensure that they are truly representative approximations.

To conclude this section on approximation concepts the occasional requirement that variables must have integer values or values picked from a discrete set is examined. These requirements considerably complicate the solution of problems. A useful way of handling them is first of all to ignore them and solve the problem using continuous variables. The solution of this problem will then be in the approximate area of the solution for integer or discrete variables in most cases. Pathological examples can be constructed mathematically in which the integer/discrete optimum is completely different from the continuous optimum but this does not occur in real-world structural optimization. Having found a continuous optimum integer programming may be used to find the integer/discrete optimum in that region, or, as is perhaps more appropriate, the designer in his analysis, checking and modification can round the continuous solution to suitable discrete values. This continuous approximation to discrete functions is most useful by virtue of the fact that it has reduced the area of search for a discrete optimum to a small region around the continuous optimum. Had this discrete search been made over the whole feasible design space it would be a very lengthy procedure.

In the field of steel civil engineering structures geometric similarity may be used to construct continuous functions representing the section properties of rolled steel sections which are available only in a range of discrete sizes. Consider, for example, the range of available wide flange beams. Select a characteristic cross-section parameter such as section area, A. The assumption is now made that all beams in the range of available sizes are geometrically similar in shape but differ only in scale. Then with this assumption it can be shown that the section modulus Z of a beam in this range is related to its area A by the relationship:

$$Z = C_1 A^{\frac{3}{2}} \qquad (2)$$

Similarly the moment of inertia, I and the depth d are given by:

$$\left.\begin{aligned} I &= C_2 A^2 \\ d &= C_3 A^{\frac{1}{2}} \end{aligned}\right\} \qquad (3)$$

and similar relationships hold for all section properties which may be related to a single characteristic parameter. The coefficients C_1, C_2, C_3, etc., are all constants for the particular range of sections and may be found by examining any discrete beam within the set.

This continuous approximation has essentially replaced the set of discrete beams by a single variable A to which all section properties are related by known functions (2),(3), etc. If variable A is used in an optimization problem the optimal value A^* will correspond very closely to perhaps two or three beams within the set from which a discrete member may be selected. The assumption of geometric similarity is not absolutely valid for all sets of available rolled sections but is sufficiently accurate for the purpose here.

Thus far this paper has examined some of the major concepts of structural optimization and some of the general techniques involved in problem formulation for real-world structures. Very often it is difficult to formulate a representative mathematical model for the design of some structures particularly in the architectural area where the component of subjectivity in design is far greater than objectivity. However, having formulated a problem in the general form of problem (1) it is necessary to be able to solve it, and so solution methods for structural optimization problems will now be considered.

4. SOLUTION METHODS IN STRUCTURAL OPTIMIZATION

It is important in any survey of methods of structural optimization to differentiate between mathematical and structural optimization techniques. Although problems of optimum structural design can be expressed in mathematical form (problem (1)), the ideal solution technique from the viewpoint of a structural engineer is quite different from what a mathematician would consider ideal. Basically the structural engineer is interested in the structure which the problem represents rather than the problem itself. He is interested in the results of the optimization rather than the means whereby they were obtained. Any optimization method for use in engineering problems must therefore be flexible enough to solve as wide a range of problems as possible - frequently problems will alter several times as new constraints and variables are introduced by the engineer to more accurately represent his real-world design problem. The method used should be robust in operation and reliable - the engineer wants useable results and is frustrated by a solution method which in operation is very sensitive to the mathematics of a problem. Any method should be comparatively easy to use and should require a minimum of pre-solution computer programming and preparation. The engineer is comparatively little bothered by the need to differentiate between global and local optimality of solutions since a great many structural optimization problems display very flat, plateau-like optima. Similarly, extreme accuracy of the solution is not necessary since the mathematical problem itself is only an approximation to a real-world structural design and the designer is aware that the optimum solution is only a guide for him and will probably require modification in accordance with factors not included in the mathematical formulation.

The mathematician, however, has a different viewpoint and is interested in the means whereby results are obtained. He is happy to develop a method which solves a very limited class of problems if it is efficient on those problems. Flexibility of a method is of comparatively little interest. Methods are often reported in the mathematical literature which have low reliability - they are efficient on some problems yet fail to produce results for purely mathematical reasons on problems which to all intents and purposes look very similar. Such methods are of little interest to an engineer who wants results all the time. By training, the mathematician concentrates upon such factors as solution accuracy, speed and accuracy of convergence to the solution, and differentiation between local and global optima.

It is not the intention to disparage the approach of the mathematician in developing optimization techniques. Without this work structural optimization would not be at its present stage of development. It should be stressed however that it is very necessary to examine all mathematical techniques carefully to determine whether they are suitable for the needs of the structural engineer. Very often the structural engineer uses relatively simple and crude techniques to solve problems, not out of ignorance of more sophisticated methods but because he can place reliance upon the results obtained. This point is often mis-understood but is an essential difference between mathematical and structural optimization.

The methods now described are those which, after careful examination and thorough testing by structural engineers, have been established as suitable for optimum structural design problems. The criticisms levelled at them are likewise based upon the performance of the methods on problems arising in the design of real-world structures. At this stage it is useful to restate the basic characteristics of most structural optimization problems which are that they are large problems with many variables and a large number of non-linear constraints.

4.1 Unconstrained Methods

It may seem odd to commence a survey of methods of structural optimization with methods for solving unconstrained problems since structural design problems almost always have a large number of constraints. However, many of the concepts of unconstrained methods are useful in constrained problems and also there are methods which transform problems having constraints to unconstrained problems. Unconstrained optimization can be formally expressed as:

$$\text{Minimize} \quad g_o\,(x_i) \qquad i = 1, \ldots, N \tag{4}$$

It is not necessary to consider maximization separately since this can be effected by minimizing the negative of $g_o\,(x_i)$. The N variables x_i represent an N-dimensional infinite design space in which all values are feasible. In order to find the minimum value of the function g_o the classical theory of optimization examines all stationary points of g_o, i.e., solves the set of equations:

$$\frac{\partial}{\partial x_i}\, g_o\,(x_i) = 0 \qquad i = 1, \ldots, N \tag{5}$$

The minimum must be one of the set of possible solutions of (5) and it can be found by substitution of all solutions of (5) into (4), the lowest result being chosen. However, there are very many circumstances under which this classical approach just does not work. Typically, it does not work if $g_o\,(x_i)$ is a non-analytic function or if some of the derivatives in (5) are discontinuous. The science of optimization theory stems from the very frequent failure of the classical approach to solve problems.

In the absence of a successful classical approach to problem (4) a logical method of finding the minimum value of g_o would seem to be to evaluate g_o at a series of trial sets of values of the variables and to numerically select the lowest result. This can be done by imposing a 'grid' over the design space and evaluating g_o numerically at each grid intersection point. However, if the likely range of each variable x_i is divided into equal divisions so as to give 10 trial values for each variable the total number of values of g_o which must be evaluated is 10^N. If there are many variables, i.e., if N is large, this number is very large and the method uses too much computer time. A random search in which trial points are selected according to a statistically random sequence is likewise inefficient because of the large number of trial evaluations of g_o which must be made even to locate a point near to the optimum.

A much better search strategy is to try to ensure that each trial evaluation of g_o is made according to a set of rules which give a good likelihood that g_o will be reduced. A vital concept in this context is that of the gradient of g_o. The gradient of g_o with respect to some variable x_i is simply the first partial derivative of g_o for x_i, i.e.,

$$\frac{\partial}{\partial x_i} g_o \,(x_i)$$

Obviously if the gradient of g_o for variable x_i is negative then if the value of x_i is increased g_o can be expected to decrease. If, at each new evaluation of g_o (x_i) all the first partial derivatives can also be evaluated then a new trial point at which the value of g_o could be expected to decrease can easily be found by either increasing or decreasing each value of x_i according to the sign of its partial derivative. Very many numerical search strategies are based upon using gradient information to produce a new search direction in which new trial evaluations may be made with maximum likelihood that g_o will decrease.

The steepest gradient method is one such strategy. The N gradients of g_o at a particular trial point represent an N-dimensional plane which is exactly tangential to the surface of g_o at the trial point. The steepest gradient method finds that direction upon the tangent plane in which the slope of the plane is maximum. Then by placing a new trial point somewhere along this direction in a decreasing sense g_o can be expected to decrease in value by more than if the trial were placed in any other direction. The steepest gradient method is a frequently used one in engineering design because of its reliability and its ease of implementation. However, though it always finds a minimum it can often converge very slowly. This is because the tangent plane is essentially a linear approximation to the surface of g_o which is exact only at the trial point. Once a new trial point is selected, even in the direction of the steepest gradient, at some distance from the original trial point the non-linearity of g_o may render the tangent plane approximation inaccurate, leading to very slow convergence.

For this reason directions other than the steepest gradient direction are often used to form the basis of a search procedure. Methods based upon conjugate directions are typical of these. Within the scope of this paper it is not possible to examine these methods in detail but their objectives should be stressed. The purpose of such methods is to improve the rate of convergence of methods like the steepest gradient which, although logical, is sometimes very slow.

All methods of search which used gradient information as well as trial evaluations are collectively termed first-order methods because they require first partial derivatives of g_o. In order to improve upon the efficiency of these methods many second-order methods have been proposed and used. These methods use the second partial derivatives of g_o, (i.e., information about the local curvature of g_o) in order to speed convergence. Once again space precludes further examination of these methods other than to comment that because they use more information about the local behaviour of g_o they are consequently more efficient. For a very readable account of many different methods for unconstrained minimization reference should be made to the work of Sargent[4].

Several comments should be made on the general applicability of zeroth-, first- and second-order methods to structural optimization. From the point of view of speed and efficiency second-order methods are obviously strong candidates. However, these methods require the prior and recurring evaluation of not only the function g_o but also all its first and second partial derivatives. This can be very time consuming. If g_o is a function of 50 variables x_i, $i = 1, \ldots 50$, then there is at each trial point a single function evaluation, 50 first partial derivative evaluations and the matrix of second partial derivatives requires the evaluation of 2500 elements. The computer time and space required for preparatory work and general housekeeping operations can be considerable in the implementation of second-order optimization methods. First-order methods also suffer from this criticism but less so. Carpenter and Smith[5] have compared the performance of zeroth-, first- and second-order methods on a selection of simple structural optimization problems by the SUMT method[8] This comparative study is valuable as it brings out very clearly the advantages and the disadvantages of each method. They conclude that on the problems they examined the first-order method of Fletcher-Powell[6] was preferable for larger problems, Newton's second-order method for small analytic problems and the behaviour of Powell's method[7] , a frequently used one, was generally poor. However, it is also fair to comment that although this sort of information is very useful it is inevitably problem-dependent. For some problems the first and second partial derivatives may be easy to obtain while for others they may be obtainable only by numerical difference techniques which can be very laborious. The choice of methods is therefore a complex one but, bearing in mind that Carpenter and Smith compared only three methods which have been available for at least ten years, their conclusions form a very useful guide.

4.2 Penalty Function Methods

The above section on unconstrained methods is necessary for an understanding of penalty function methods which solve a constrained problem by means of a sequence of unconstrained problems. Penalty function methods have been widely used in structural optimization and are among the more popular methods. Consider the equality-constrained problem:

$$\left.\begin{array}{lll} \text{Minimize} & g_o\,(x_i) & i = 1, \ldots, N \\ \text{Subject to} & g_j\,(x_i) = 0 & j = 1, \ldots, M \end{array}\right\} \quad (6)$$

The penalty function approach replaces problem (6) by the unconstrained problem:

$$\text{Minimize } F = g_o\,(x_i) + \sum_{j=1}^{M} P_j \left[g_j\,(x_i) \right]^2 \qquad i = 1, \ldots, N \qquad (7)$$

in which values of P_j, $j = 1, \ldots, M$, are positive constants. The function F is therefore composed of the original objective function g_o plus the value of each constraint multiplied by a penalty factor P_j. Starting with some known set of factors P_j, F is minimized using unconstrained techniques. All values of P_j are then considerably increased and another unconstrained minimization of F is performed. This process continues for increasing values of the penalty factors and has the effect of forcing each of the constraint functions g_j towards the value zero. Thus as values of P_j are increased the results of the sequence of unconstrained minimizations of F tend towards the solution of problem (6).

Penalty function methods are also applicable to inequality constrained problems. If the constraints g_j in problem (6) are written as:

$$g_j \ (x_i) \leqslant 0 \qquad j = 1, \ldots, M \qquad (8)$$

a suitable interior penalty function problem is:

$$\text{Minimize } F = g_o \ (x_i) - P \sum_{j=1}^{M} [g_j \ (x_i)]^{-1} \qquad i = 1, \ldots, N \qquad (9)$$

where P is a positive penalty factor. Starting with a large value of the penalty factor P an unconstrained search is carried out from a feasible starting point, (i.e., values of x_i which do not violate any of the constraints - hence an interior or feasible point). The solution of this search cannot be at a point which causes any constraint value g_j to be zero otherwise F would be infinite. This time the effect of P is to keep the solution away from the constraint boundaries. Further unconstrained searches are carried out using a sequence of decreasing values of P thus the process can progressively approach any constraint boundary, where $g_j = 0$, if it wishes or can remain feasible, $g_j < 0$, if this is advantageous. The results of the sequence of problems thus converges to the minimum of g_o with constraints given by (8).

Much work has been published on penalty function methods and both interior and exterior methods (the search is always in the infeasible region) have been widely studied. A popular method is the SUMT method of Fiacco and McCormick[8] to which reference should be made. Lootsma[9] has given a comprehensive review of the topic and its use in structural optimization has been championed by Fox[10] and Moe[11] in particular.

The chief disadvantage of the approach is that although it converts constrained optimization problems to the much simpler unconstrained form, it requires a considerable amount of time to solve the unconstrained problem many times. Furthermore the composite unconstrained objective function F contains all the constraint functions g_j and since we have seen that structural optimization problems usually have many constraints F can be very large. In solution of the unconstrained minimization if first-order or second-order methods as described in section 4.1 are used, partial derivatives of all the constraint functions g_j must be evaluated since they appear in F. Thus considerable time is necessary for the evaluation of derivatives and for this reason the penalty function approach cannot really be deemed suitable for large structural optimization problems. It has been used very effectively on the detailed design of structural elements or components such as beams, plates, panels, etc., where the number of variables is perhaps a maximum of 15 and the number of constraints is of the same order. For more complex problems its efficiency can sometimes be rather poor.

4.3 Constrained Numerical Search

In section 4.1 methods of unconstrained numerical search were considered. Here the feasible region is infinite. When constraints are present, however, they limit the feasible region which is hedged around by constraints which must not be violated. The only way of knowing whether a particular trial point is or is not feasible is to evaluate all constraints at that point and check them for violation. If methods of gradient search are to be used it is likewise vital to know if a particular search direction points into an infeasible region. This too involves checking all constraints and the derivatives of constraints. It is obvious, therefore, that the presence of many non-linear constraints which is characteristic of structural optimization problems causes considerable difficulties for any numerical optimization method. The success and efficiency of any search technique depends upon the nature of the constraints.

The simplest case is where all constraints are linear equalities. Here as many variables may be eliminated (expressed in terms of other variables) as there are equality constraints and the problem is then reduced to an unconstrained one in a reduced number of variables. The linear equality constraints positively help the solution process. When the problem is constrained by linear inequalities slack variables may be added to convert the constraints to equalities and the solution can then be carried out as above by unconstrained methods. Linear constraints can therefore be handled quite easily and efficiently.

In structural optimization the constraints are usually and unfortunately non-linear inequalities. It is the author's opinion that no numerical search method for non-linear inequality constraints has yet been developed which can be advocated on grounds of reliability and efficiency as suitable for anything but the smallest structural optimization problems, i.e., about 5 - 10 variables and a similar number of constraints. It is almost always possible to solve problems more efficiently by the penalty function approach of section 4.2 or by the methods to be outlined in the following sections. Many methods have been proposed and they founder generally on the need for many trials and derivative evaluations. These are necessary because of the difficulty experienced in locating and following a non-linear boundary in N-dimensional space. The only direct search method for non-linear constraints which could possibly be an exception is that based upon the Simplex method of Nelder and Mead[12] which is a zeroth-order method making trial evaluations at the vertices of a N-dimensional regular figure which 'spins' through the feasible design space.

4.4 Linear Programming

Throughout this paper it has been emphasized that structural optimization problems are generally highly non-linear. There is, however, a major exception to this generalization which arises in the optimum plastic design of structures. Consider the beam/column framework of a multistorey building such as that shown in Fig. 2. The design of such a framework on a fully-plastic basis consists of finding a set of fully-plastic moments M_p for all members of the framework so that a prescribed factor of safety against collapse is achieved. The optimum design is one in which the set of M_p values also minimizes the weight or cost of the frame. For a frame of given layout member lengths are known and it is possible to approximate the cost function for the frame as a linear function of the M_p values of all the members. It should be noted that this is only an approximation but is a very reasonable one to make in order to solve the design problem which is in category 3 of the hierarchy described in section 2.

The constraints upon the problem are those of structural mechanics: It is necessary to ensure that in any possible collapse mode the work done by the factored applied loads does not exceed the energy capacity of the rotations at plastic hinges in the frame. This requirement leads to constraints in which linear functions of the M_p values are bounded above by a known set of constants. For completeness there should be one linear constraint for each possible collapse mode. In this problem, then, both the objective function and all the constraint functions are linear and linear programming may be used to find the optimal set of M_p values.

Optimum fully-plastic design has received much attention in the past since linear programming may often be used for which efficient and reliable computer package programs are available. However, the objective function is not truly linear and from a structural point of view the constraints are incomplete since several non-linear effects have to be omitted. For instance, elastic instability of the frame, the reduction of plastic moment capacity due to axial load, and change of geometry effects cannot be included. There are also considerable problems involved in ensuring that the actual collapse mode of the frame is present among the constraints. Much research effort has been devoted to overcoming some of these difficulties which is ample testimony to the ease of use and popularity of linear programming.

Linear programming deals with problems similar to problem (1) in which all functions g_j, $j = 0, \ldots, M$ are linear. Almost any textbook on optimization or operational research gives several solution techniques so no specific reference is made here. Several important features of linear programming can be mentioned, however. The first is that very large problems can be solved very efficiently. Many thousands of variables and constraints can be handled effectively. It is perhaps for this reason alone that linear programming is popular as a sequential method, (see section 4.5), for structural optimization. It is very worthwhile trying to force a problem into a linear format if possible because very large problems can then be solved. Secondly the duality theory of linear programming has the advantage that it gives insight into the nature of the problem which normally remains obscure. Linear duality may be summarized as follows:

<u>Primal Problem</u>

Minimize $W = \sum_{i=1}^{N} C_i \, x_i$

Subject to

$$\sum_{i=1}^{N} a_{ji} \, x_i \geq b_j \qquad j = 1, \ldots, M$$

$$x_i \geq 0 \qquad i = 1, \ldots, N$$

<u>Dual Problem</u>

Maximize $Y = \sum_{j=1}^{M} b_j \, \lambda_j$

Subject to

$$\sum_{j=1}^{M} a_{ji} \, \lambda_j \leq C_i \qquad i = 1, \ldots, N$$

$$\lambda_j \leq 0 \qquad j = 1, \ldots, M$$

Thus each primal linear programming problem in variables x_i has a dual problem in variables λ_j such that the solution of either one is exactly equivalent to solution of the other. A problem with few variables and many constraints has an equivalent dual problem with many variables but few constraints, and so the easier of the two problems may be solved.

Space precludes any further consideration of duality other than to mention that the physical interpretation of dual variables for a real-world structural design problem remains a very fertile area of research. Many authors[13,14] have examined linear duality for problems arising in fully-plastic design of structures and as a result have contributed greatly to an understanding of the basic behaviour of structures. As a final advantage of linear programming it should be noted that restrictions that variables take integer values or values from a discrete set can be accommodated by linear programming in a rigorous manner. This does not generally apply to other methods, apart from Dynamic programming.

4.5 <u>Sequential Linear Programming</u>

Because linear programming is an efficient and reliable technique which can solve very large problems it has been used widely in a sequential manner for the solution of non-linear problems. The way this is done is as follows. Consider problem (1) with all functions g_j, $j = 0, \ldots, M$ being non-linear. Take a feasible trial point $\overline{x}_i$, $i = 1, \ldots, N$, and evaluate at this point the values of all functions g_j and the values of all first partial derivatives of all functions g_j. Each of the functions g_j may then be replaced by a linear approximation which has the same value and gradients at the trial point. The approximating functions can be easily derived from a first-order Taylor series expansion about the trial point. The resulting approximate problem is now a linear programming one and may be solved as such. The result of this linear programming problem then gives a new trial point at which a new approximating linear problem can be constructed. The solution of problem (I) is then approached by means of a sequence of linear programs.

The sequential linear programming method can be criticised in several ways. Firstly, there is no proof that the sequence of linear approximations will converge to the optimum of problem (I). Conceptually, convergence is often assumed but if the degree of non-linearity in the original problem is high convergence of the sequence may be very slow. Also, a linear approximation to a non-linear problem may be very inaccurate and the results of each linear program may be highly infeasible for the original non-linear problem. In order to ameliorate this move limits are often used which add extra constraints to prevent the linear search from going too far into infeasible regions. This increases the number of cycles of iteration and has the effect of imposing convergence to a point which is not necessarily the optimum of problem (I). The use of linear programming implies that the optimum will always be found at a vertex of the linear constraints but this is not necessarily true of general non-linear problems.

However, despite these criticisms sequential linear programming remains a very popular method of structural optimization and although its performance sometimes leaves much to be desired it has been successfully used on a very wide range of large and complex structural design problems. Pope[15] has given a good description of sequential linear programming in AGARDograph 149 which is entitled 'Structural Design Applications of Mathematical Programming Techniques' and gives excellent background material in the area of structural optimization. A big advantage of the SLP method is that it can tackle large and complex problems unsuited to any other technique. It has frequently been linked to finite element analysis programs to give an iterative optimum design capability. In such cases considerable amounts of computer time are necessary and the final results are sometimes only approximate optima but this method is the only way in which complex problems can be solved. Examples in the field of structural design are many but two may be referred to as they typify the performance of the method on very complex problems[16,17].

4.6 Dynamic Programming

Dynamic programming solves a very special class of problems in which the objective is to extremize the performance of a serial system.

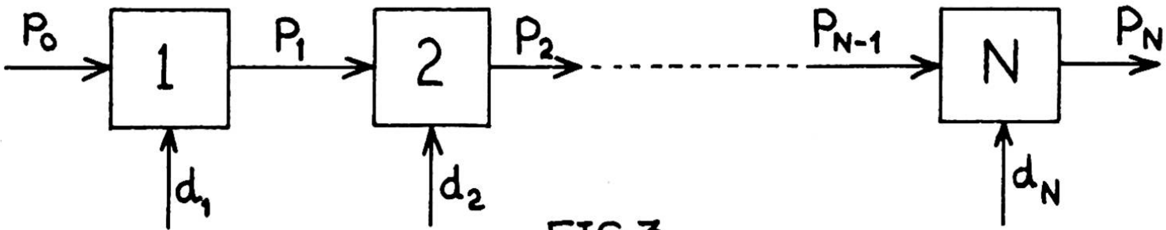


FIG.3

In Fig. 3 each box represents a stage in the serial system. Consider box 1. Input to this stage is the value of the performance p_o. By giving different values to the decision variable d_1 the output performance p_1 of stage 1 is varied. This output p_1 of stage 1 is the input to stage 2 and the system performance at the end of stage 2, p_2, is modified from the input value p_1 by decision variable d_2. By the final stage N, the final performance criterion p_N is therefore a function of the initial performance p_o and the N decision variables

d_1 ..., d_N. The objective is to find values for p_O and all the N decision variables which extremize the final performance p_N. Dynamic programming is a very efficient technique for solving such problems and was developed by Bellman[18].

The dynamic programming method itself is not studied here but some characteristics of the problem should be noted. First of all it is necessary to have a serial system such as shown in Fig. 3 in which decisions taken at a particular stage affect only performance after that stage and not before it. No 'looping-back' is possible. Although Fig. 3 shows only one decision variable at each stage it is possible, though more expensive in time, to have multiple decisions at each stage. The reason that dynamic programming is mentioned in connection with structural optimization is that it is very rapid and efficient in solving problems which fall into the suitable class. Unfortunately very few structural optimization problems have the necessary sequential decision characteristics. Palmer[19,20] has applied the method to optimizing the geometry of transmission towers and frameworks with some success. As a general comment upon the dynamic programming method for structural optimization it can be said that it works very well indeed on suitable problems but few structural problems are suitable.

4.7 Geometric Programming

Geometric programming is perhaps the most recently developed mathematical programming method to appear on the structural optimization scene. Like linear programming and dynamic programming, geometric programming solves a particular class of problems but unlike these other methods the class of suitable problems is quite large and many optimum structural design problems are suitable. Geometric programming in its simplest form solves the problem:

$$\left.\begin{array}{lll} \text{Minimize} & g_o\,(x_i) & i = 1, \ldots, N \\ \text{Subject to} & g_j\,(x_i) \leq 1 & j = 1, \ldots, M \\ & x_i \geq 0 & i = 1, \ldots, N \end{array}\right\} \quad (10)$$

with the restrictions on the mathematical form of the functions g_j that:

$$g_j\,(x_i) \equiv \sum_{t=1}^{T_o} C_{jt} \prod_{i=1}^{N} x^{a_{jti}} \qquad j = 0, \ldots, M \qquad (11)$$

$$\text{and } C_{jt} > 0 \qquad \text{for all } j,\ t \qquad (12)$$

The class of suitable problems is therefore governed by the requirement that all functions be <u>sums</u> of terms, each term involving <u>products</u> of variables raised to known powers. Generally all constraints are non-linear inequalities and such problems can be very difficult to solve directly. Geometric programming does not attempt direct solution but uses theorems of geometric duality, described by Duffin et al[21] to construct an equivalent dual problem. Using dual variables δ the geometric dual problem exactly equivalent to (10), (11) and 12) is:

$$\text{Maximize } V\,(\delta) = \prod_{j=0}^{M} \prod_{t=1}^{T_j} \left(\frac{C_{jt}\,\lambda_j}{\delta_{jt}}\right)^{\delta_{jt}} \qquad \Bigg\}$$

Subject to the constraints $\sum_{t=1}^{T_o} \delta_{ot} = 1$

$$\sum_{j=0}^{M} \sum_{t=1}^{T_j} a_{jti}\, \delta_{jt} = 0 \qquad i = 1, \ldots, N \qquad (13)$$

in which $\lambda_o = 1$

$$\lambda_j = \sum_{t=1}^{T_j} \delta_{jt} \qquad j = 1, \ldots, M$$

$$\delta_{jt} \geq 0 \qquad \text{all } j, t$$

The important feature of the geometric dual problem (13) is that the constraints are linear equalities. As was mentioned in section 4.3 linear equality constraints actually assist in maximizing V (δ) and methods of unconstrained search can be adapted to solve the dual problem numerically. The geometric programming method therefore uses duality theorems to convert problems of a difficult type to solve directly into problems with linear equality constraints which are much easier to solve. Since the appearance of Duffin, Peterson and Zener's book[21] in 1967 the method has been considerably extended and requirement (12) that all coefficients must be positive has now been removed in more recent versions of the method. Also constraints of reversed sign can be handled. Templeman and Winterbottom[22] have summarized these recent developments in a paper which describes a computer program for geometric programming and its application to structural design.

Advantages of the method are that it is a non-linear one which works effectively on highly non-linear problems. The class of suitable problems is clearly defined and is as easy to recognize as linear programming. A standard format of problem input is available which makes solution by means of computer package programs relatively easy. It has been shown that very many structural optimization problems are naturally suitable for solution by geometric programming. The detailed design of almost all types of structural elements (category 4 of the hierarchy described in section 2) can be expressed as geometric programming problems. Equations (2) and (3) show that the physical properties of a beam or column can be related to a single variable, A, raised to different powers. Terms such as these typify category 4 problems and they are in the form of (11) which is ideal for geometric programming. Templeman[23] gives several examples of the optimum design of structural components such as beams, corrugated plates and integrally-stiffened compression panels and also shows how geometric approximations may be made to more general problems in structural optimization. Non-standard problems can then be solved by a converging sequence of geometric programming problems in a conceptually similar way to sequential linear programming as described in section 4.5. This has the potential advantage that more representative non-linear approximations rather than linear ones are made to non-linear problems. A final advantage of the method is that the dual problem sheds new light on the optimum design and much insight can be gained by attempting to interpret physically the dual of a structural design problem. This point was made previously in connection with linear duality and it seems likely that research now being carried out in several centres into interpretation of primal/dual systems could in the future be of great benefit, leading to new structural optimization techniques and enhanced understanding of the design process and of structural behaviour.

Finally, the disadvantages of the geometric programming method should be mentioned. Firstly, it is the most difficult to understand of the methods described here. Secondly, if a computer package is not readily available to a prospective user of geometric programming then the effort involved in programming the method is very large. At present there are only a few suitable computer programs available. Thirdly the method is still relatively in its infancy, its use is not widespread and much of its great potential is still to be realized. Finally, the performance of the method on very large non-linear problems is dominated rather by computer storage and hence also by run-time. Factual evidence of this is as yet small but it may be that there is a fairly high limit upon the size of geometric programming problems which can be solved within normal economical limits.

5. OPTIMALITY CRITERION METHODS

All the methods described in section 4 attempt to solve structural optimization problems of the general form of problem (I) by mathematical and numerical search methods. The philosophy adopted is that nothing is assumed about the location or nature of the optimum. The optimum is reached by some purely numerical search which is based upon the mathematical form of problem (I) rather than the real-world structural design which problem (I) represents. The structural engineer, however, sometimes feels that he knows much more about the optimum structure than is present in its mathematical equivalent and that this knowledge might be useful in deriving a search method based upon structural principles rather than upon a mathematical abstraction. The term optimality criterion methods covers such approaches. Optimality criterion methods solve problems such as problem (I) by search methods so strictly this section could be numbered 4.8 and considered with all the other methods. The concept is so different, however, as to warrant a separation from the other methods.

Optimality criterion methods are absolutely problem-dependent and a particular criterion applies only to the optimum design of a particular type of structure under very specific conditions. The philosophy of the optimality criterion approach is first of all to investigate the nature of optimum structures of some specific type to try to establish a condition or set of conditions satisfied only by the optimum structure and which are not satisfied by any other, non-optimal design. For example such conditions might be that for a particular type of structure under restrictions on stresses and displacements the optimum structure always has a recognizable distribution of some form of energy among its components and that this distribution is peculiar only to the optimum design. Having found some structural criterion of optimality it is then necessary to devise some iterative algorithm which, starting with a non-optimal structure, will successively redesign the structure so that a structure which satisfies the optimality criterion and hence is optimal will be found.

For particular classes of structures the optimality criterion approach solves the mathematical problem (I) very indirectly by completely replacing the problem by an analogous one of iteratively redesigning a structure so that it satisfies some pre-established criterion of optimality. In a sense optimality criterion methods are dual methods in that problem (I) is solved by solving a completely different but equivalent 'dual' problem. The nature of this duality between mathematical programming and optimality criterion methods is almost totally unexplored but holds out considerable promise for further research work.

In developing an optimality criterion method for a particular class of structures there are two distinct phases. First of all a relevant and unique criterion of optimality for the class of structures must be found. This is generally very difficult to do since such criteria are rarely obvious. Had optimality criteria been obvious then mathematical programming would never have been necessary for optimum structural design. It is the requirement that the criterion be unique to the optimum structure which is perhaps most difficult to satisfy and indeed at the present time the technical literature holds examples of many so-called optimality criteria which are not optimal at all in that they can also be satisfied by non-optimal structures. The second phase is that of developing a recursion relationship which will produce an iterative redesign algorithm so that the optimality criterion can be satisfied. This too can frequently be difficult to develop and it runs the risk like any numerical search method of being unwieldy to operate or slow to converge. The optimality criterion approach is a logical one which has great appeal to structural engineers since it is based upon structural rather than mathematical principles. It holds great promise for the future although it can never entirely replace mathematical programming methods but, like some of the methods described earlier, it is still in its infancy and it still has to realize much of its potential.

At present very few rigorous, well-tested optimality criterion methods exist although many have been proposed and the range of optimum structural design problems which they cover is small. Generally they apply to overall sizing of structural members in a multimember system such as a truss or frame, i.e., category 3 problems in the hierarchy of section 2. Some authors, notably Prager[24], have developed optimality criteria for structural components such as beams, sandwich plates, etc. For such structures the optimality criterion generally is concerned with energy distributions in the parts of the structure. Prager has concluded that volume integral of energy density in each part is proportional to the volume of the part in many cases with single constraints. When multiple constraints are present the optimality criterion becomes more complicated but this energy distribution pattern is still an optimality criterion although in a modified form. Variations upon this energy-density optimality criterion have been proven to be applicable to truss structures, notably by Venkayya[25] and many others. Indeed many types of structures may be designed by optimality criteria methods with constraints upon stresses or displacements or dynamic stiffness, and the big potential advantage which the optimality criterion approach has is that very large problems can be designed this way, given a suitable criterion and algorithm, whereas mathematical programming methods are often suitable only for relatively smaller structures. A major difficulty is encountered when structures are to be designed subject to multiple constraints of different types. For example, a truss may be required to satisfy both stress constraints for which a criterion is available and multiple displacement constraints for which another criterion is known. Both criteria are not usually satisfied simultaneously and although a composite optimality criterion can be devised it is considerably more difficult to devise a rapid redesign algorithm and the solution process can be very slow.

There are therefore many difficulties associated with the optimality criterion approach and these are not always immediately obvious. The idea of developing structurally based rather than mathematically based optimization methods is, however, very appealing and holds out great hope for the future but much more work remains to be done in order to realize this potential.

6. CONCLUSIONS

Structural optimization is at present a thriving area of research and development. The philosophy is so very obviously right since the structural engineer has historically been guided by the need and desire to produce structures which are in some respect 'better' or more efficient than those which have gone before. The electronic computer has enabled design and optimum design to be put on a more formal and rigorous basis and is the means by which the goals of structural optimization may be achieved. However, it is fair comment to say that the actual methods of optimization at present available are not entirely adequate for very many of the structural optimization problems which surround us today. Mathematical programming methods can sometimes be laborious in operation and are often restricted to only small to medium sized problems. Some methods offer more promise for the future than others and mathematical programming is still very much alive. In the author's opinion better methods will be developed in the future perhaps based upon duality. Optimality criterion methods have great appeal to the structural engineer yet at present they too are in their infancy and require much more development.

In section 2 of this paper a hierarchy of structural optimization problems was discussed. Research up to the present time has tended to be concentrated mainly in categories 3 and 4 of this hierarchy - the easier problems. Categories 1 and 2 are as yet little-explored and the potential rewards offered for methods of solving such problems are very great. By about 1970 a watershed had been reached in structural optimization. Prior to this date research had concentrated upon marrying existing solution methods to structural design problems and it became evident that the marriage was only partially successful. Very many pressing problems remained to be solved. Since 1970 progress in structural optimization has been along new lines of approach. The simpler problems are now things of the past and only the harder ones remain. It is significant that interest in these harder problems of structural optimization is unabated. Over the last five years many new lines of approach have been opened up and although progress has often been slow the full potential of the new methods can now be clearly seen as a future goal. To achieve this potential is the object of structural optimization today.

7. REFERENCES

1. Schmit, L. A. - 'Structural design by systematic synthesis', Proc. 2nd Conf. on Electronic Computation, ASCE, 1960.
2. Michell, A. G. M. - 'The limit of economy of material in frame-structures', Phil. Mag., Vol. 8, No. 47, London, 1904.
3. Hemp, W. S. - 'Optimum structures', Oxford University Press, 1973.
4. Sargent, R. W. H. - 'Minimization without constraints', in 'Optimization and Design', Avriel, M., Rijckaert, M. J., Wilde, D. J. (Eds), Prentice-Hall, Englewood Cliffs, N.J., 1973.
5. Carpenter, W. C. and Smith E. A. - 'Computational efficiency in structural optimization', Engineering Optimization, Vol. 1, No. 3, 1975.
6. Fletcher, R. and Powell, M. J. D. - 'A rapidly convergent descent method for minimization', Computer Jnl., Vol. 6, No. 2, 1963.
7. Powell, M. J. D. - 'An efficient method for finding the minimum of a function of several variables without calculating derivatives', Computer Jnl., Vol. 7, No. 2, 1964.
8. Fiacco, A. V. and McCormick, G. P. - 'Nonlinear Programming. Sequential Unconstrained Minimization Techniques', John Wiley, New York, 1968.
9. Lootsma, F. A. - 'A survey of methods for solving constrained minimization problems via unconstrained minimization', in 'Optimization and Design', Avriel, M., Rijckaert, M. J., Wilde, D. J. (Eds), Prentice Hall, Englewood Cliffs, N.J., 1973.
10. Fox, R. L. - 'Optimization methods for engineering design', Addison-Wesley, Reading, Mass., 1971.
11. Moe, J. - 'Penalty function methods', Symp. on Optimization and Automated Design of Structures, Report No. SK/M21, Div. of Ship Structures, Tech. U. of Norway, Trondheim, 1972.
12. Nelder, J. A. and Mead, R. - 'A simplex method for function minimization', Computer Jnl., Vol. 7, No. 4, 1965.
13. Munro, J. and Smith, D. L. - 'Linear programming duality in plastic analysis and synthesis', Proc. Int. Symp. Computer-aided Structural Design, Warwick,1972.
14. Foulkes, J. D. - 'Linear programming and structural design', Proc. 2nd Symp. Linear Programming, Washington, D.C., 1955.

15. Pope, G. G. - 'Sequence of linear programs' in 'Structural Design Applications of Mathematical Programming Techniques', Pope, G. G. and Schmit, L. A. (eds), AGARD-AG-149, 1971.
16. Smith, G. K. and Woodhead, R. G. - 'An optimal design scheme with application to tanker transverse structure', Engineering Optimization, Vol. 1, No. 2, 1974.
17. Wills, J. - 'A mathematical optimization procedure and its application to the design of bridge structures', Dept. of the Environment, Transport and Road Research Laboratory, Report LR.555, Crowthorne, 1973.
18. Bellman, R. E. - 'Dynamic programming', Princeton Univ. Press, 1957.
19. Palmer, A. C. - 'Optimum structural design by dynamic programming', Jnl., ASCE, No. ST8, 1968.
20. Sheppard, D. J. and Palmer, A. C. - 'Optimal design of transmission towers by dynamic programming, 'Computers and Structures , Vol. 2, 1972.
21. Duffin, R. J., Peterson, E. L. and Zener, C. M. - 'Geometric programming. Theory and applications', John Wiley, New York, 1967.
22. Templeman, A. B. and Winterbottom, S. K. - 'Structural design applications of geometric programming,' AGARD-CP-123, 1973.
23. Templeman, A. B. - 'The use of geometric programming methods for structural optimization', AGARD Lecture Series No. 70, 1974.
24. Prager, W. - 'Optimality criteria in structural design', AGARD-R-589, 1971.
25. Venkayya, V. B., Khot, N. S. and Berke, L. - 'Application of optimality criteria approaches to automated design of large practical structures', AGARD-CP-123, 1973.

SUMMARY

The paper considers the economic objectives of structural optimization and shows that it has put design on a formal and rigorous basis. The size and type of structural optimization problems is then examined and a hierarchy of problem categories is discussed. Approximation methods are considered which enable complex structural problems to be posed concisely. Seven mathematical programming methods are described and critically discussed in the context of a specification for a suitable optimization technique for engineering use. Optimality criterion methods are then examined and the paper concludes that presently available structural optimization techniques have yet to achieve their full potential.

RESUME

L'article traite des buts économiques de l'optimisation structurale et montre que le dimensionnement est posé sur une base formelle et rigoureuse. Le type et l'importance des problèmes d'optimisation structurale sont examinés; une hiérarchie des catégories de problèmes est discutée. Des méthodes d'approximation permettent de poser des problèmes structuraux complexes de façon concise. Sept méthodes de programmation mathématique sont présentées et comparées dans le cadre de directives pour une technique d'optimisation appropriée à l'usage de l'ingénieur. Des critères de méthodes d'optimisation sont discutés. L'article conclut que les techniques actuelles d'optimisation structurale peuvent encore être améliorées.

ZUSAMMENFASSUNG

Der Beitrag behandelt die wirtschaftlichen Aspekte bei der Optimierung von Tragwerken und zeigt, dass die Bemessung auf einer formalen und strengen Grundlage beruht. Grösse und Typen baulicher Optimierungsprobleme werden untersucht und eine Rangordnung der Probleme diskutiert. Es werden Näherungsmethoden untersucht, welche gestatten, komplexe bauliche Probleme rasch zu lösen. Sieben mathematische Programmierungsmethoden werden beschrieben und hinsichtlich des Anwendungsbereiches kritisch verglichen. Sodann werden verschiedene Optimierungskriterien untersucht; der Beitrag schliesst mit der Feststellung, dass die gegenwärtig erzielte Optimierungstechnik ihre vollen Möglichkeiten noch nicht erreicht hat.

System and Geometrical Optimization for Linear and Non-Linear Structural Behaviour

Optimisation des systèmes et des dimensions pour des comportements structuraux linéaires et non-linéaires

Optimierung der Systeme und der Abmessungen bei linearem und nicht-linearem Verhalten des Tragwerkes

FRED MOSES
Professor of Civil Engineering
Case Western Reserve University
Cleveland, Ohio, USA

1. INTRODUCTION

Many discussions in the I.A.B.S.E. proceedings have considered structural optimization. For example, Professor Courbon defined optimization as designing and construction a structure at the lowest cost, with the object of fulfilling a well defined purpose.(1) In particular, cost consideration must also be given to safety, service life, maintenance and future adaptability. Since all of research and practice in structural engineering is aimed towards such a goal the activity known as structural optimization must be defined in a unique way. That is, the development and application of a priori directed and automated techniques for improving designs within well defined cost contexts and recognized constraints. Within this definition, therefore, questions of design creativity and ingenuity are put aside in favor of quantitative comparisons among a vast array of acceptable yet competing designs. Thus, in much the same way as computer methods of structural analysis, the techniques of structural optimization become an aid to the designer for rapidly proportioning structural details and evaluating design alternatives to obtain the best design among given choices. In this way, when the engineer arrives at comparing quite different conceptual designs for the same application he is fairly certain of intelligently comparing these alternatives and not unfavorably biasing one alternative by poor proportioning of its details.

In the field of structural optimization the computer becomes central as a tool for searching and sorting through the similar design concepts and proportioning the element details for the most economical design. Naturally, it arrives at a design which the engineer could equally have obtained if he were prepared to invest the time and money to directly search among the design alternatives. The principal advantage of the optimization methods is therefore its saving in design time and cost. A further consideration with the expanding usage of computer analysis programs to calculate structural behavior is that an optimization program can eliminate much of the input-output and the costly data handling effort. Since the methods result in a proportioned structure which satisfies applicable codes it becomes unnecessary to review the vast amounts of structural analysis output. Instead, the optimization program produces design details as an output and in some cases could be further programmed to produce drawings and material and fabrication specifications.

2. RAISING THE DESIGN HIERARCHY

Since structural optimization over the last decade and a half has successfully concerned itself with computer aided proportioning of design details it is easy to ignore further applications. In fact, some recent work has shown the possiblity of introducing into the automated computer procedures design variables which had previously been thought to be either in the realm of creative decisions or else difficult to program for computer selection. These additional design variables have described geometry and shape of structures, material choice, complete building design including comparison of basically different element types and design selections including overall fabrication costs and material availability. Before embarking on a description of several such examples it is worth considering from this framework the historical developments of optimization applications.

There is in this regard an analogy between computer developments in both structural analysis and optimization. When digital computers first became available civil engineers who were among its early intensive users simply programmed classical methods of analysis such as slope deflection and moment distribution. In a similar way the first structural optimization applications were programs using such well known iterative design methods as structural index, stress-ratio, fully stress and other optimality criteria.(2)

A second development of computer applications were matrix analysis for specific structural types such as trusses, frames and grillages. Paralleling this was specialized optimization programs for these same structural types using methods such as gradient directions or other heuristic design search procedures.(3) The current stage of development includes general purpose computer programs usually available from machine manufacturers or time-share agencies. In analysis this often means finite-element packages for linear and non-linear behavior. The same evolution for design has been program packages each able to handle a wide array of different structural elements and systems. It is important to investigate in detail these current developments in optimization since they include the methodology for extending applications in both structural system and geometry optimization.

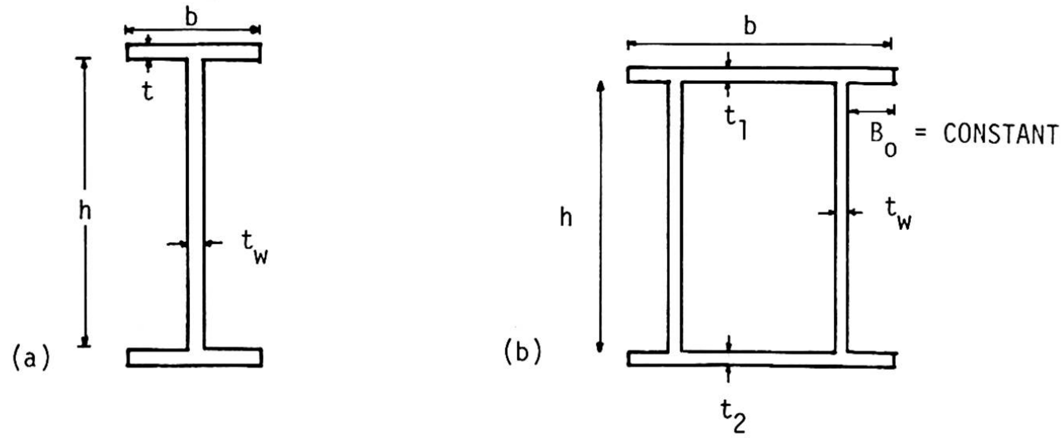
3. GENERAL PURPOSE OPTIMIZATION PROGRAMS

To explore the available methodology for structural optimization the author prefers to divide iterative structural decision problems into three categories. These are: 1) element design, 2) interconnected structural systems and 3) discrete decision variables. Each of these categories will now be considered. It is recognized that other areas could be added particularly as we move further into conceptual and creative decision variables but the three categories will suffice to cover the needs of a computer optimization library for a typical design practice.

Element Design - This design problem is characterized by a well defined code of practice for the constraints and a relatively direct method of calculating the element loading such as moment, shear, torsion and axial load. Some examples of application are shown in Figure 1 including welded wide flange, unsymmetrical box girder and prestressed beams.(4-6) Design variables are typically depths, thickness, shape, reinforcement ratios and other design details which are often part of the tedious aspects of design and for which economic selection rules are not always available. Other published examples include welded columns, gabled frame, stiffened ship plates, shear walls, prestressed plates and reinforced concrete beams.(7-11) These examples typically have a small number of independent design parameters (say less than 10) but relatively complex functional code constraints such as allowable tension, compressive and shear stresses, buckling and displacement constraints. In one example using the A.I.S.C. code the constraints were due to lateral buckling and were discontinuous representing a transition from elastic to inelastic behavior.(4) In some applications it becomes necessary to repeat the design for a large number of different elements. For example, in the case of box girder sections a particular overhead crane manufacturer using such sections needed over 5000 specified designs.(5) This obviously required an efficient design program.

Several computer packages have been developed to automate the solution to element optimization. The author prefers to use programs based on the penalty technique which combine the criteria function (cost or weight) and the constraint into a single expression to be minimized.(12) This transforms the more difficult non-linear programming problem with cost and constraints into a more tractable unconstrained minimization for which many straightforward solution methods are available. Other methods have also been successfully used on some problems including geometrical programming and linear programming.(13) As was mentioned earlier, element design is also characterized by a direct calculation of the forces on the elements. This may either be for statically determinate structures or even for complex frameworks in which a matrix structural analysis is used to solve for element forces. It is assumed in the latter case that the element size does not affect the force distribution within the structure; if this effect is significant then several cycles of iteration may be necessary to converge both the force distribution and element design. Since an element design optimization usually involves comparing many alternatives it would be computationally difficult to repeat the force analysis each time a design parameter were changed.

System Optimization - This refers to structures where there is a major physical interaction between different elements or there exists design constraints based on total structural behavior such as stability, stiffness, vibration and dynamic responses. Furthermore, system optimization arises when some of the design variables relate to more then



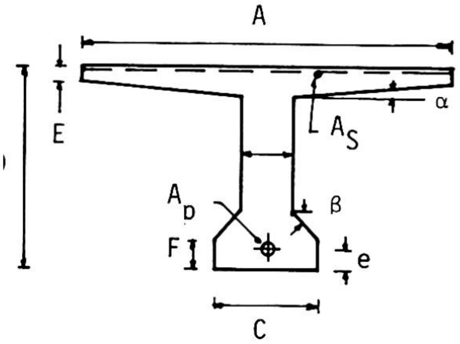
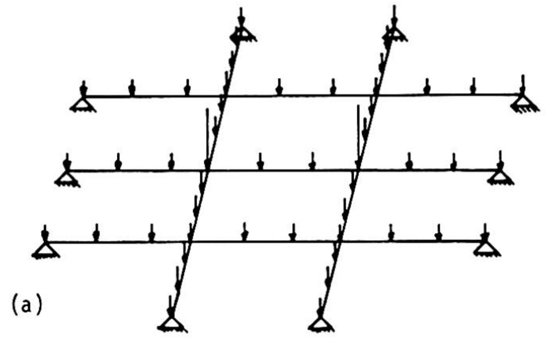
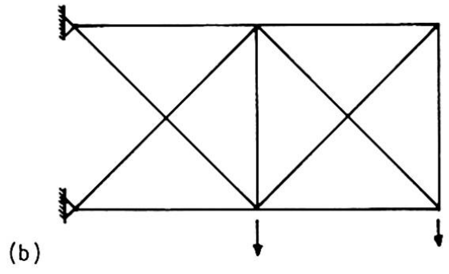


FIGURE 1: EXAMPLES OF ELEMENT DESIGN OPTIMIZATION

(a) Wide flange beam four variables, ref. 4
(b) Welded box girder, 5 variables, ref. 5
(c) Prestressed concrete beam, eleven variables, ref. 6





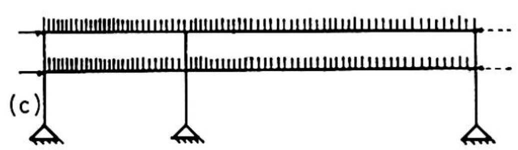


FIGURE 2: EXAMPLES OF STRUCTURAL SYSTEM OPTIMIZATION

(a) Flat orthogonal normally loaded grillage, ref. 14 variables - member sizes,
(b) Planar truss, variables - member areas, ref. 15
(c) Planar frame, variables - member sizes, ref. 16

one element such as geometry, material or topology. In these cases, the force distribution invariably requires a matrix or finite element analysis. Computer costs usually restrict the number of possible trials that the computer can sequentially examine as it searches for the optimum to often under 10 trials. Some cases that have been optimized include foundation or ship grillages, trusses and frames and are illustrated in Figure 2.(14-16) The most efficient search techniques for systems usually utilize some form of linear programming steps since in this way large changes in design may be made with only a single analysis interaction.(16,17)

Discrete Selection Parameters - This category of optimization is considerably more complex then either the two first categories. Discrete design variables are not always clearly defined or constant in number and the cost function may be extremely complex and discontinuous. The discrete nature of variables often requires heuristic or intuitive search methods of solution which make general purpose programs irrelevant. One method however, which has solved a variety of such problems is the dynamic programming technique which is easy to program if the basic problem formulation meets its definition.(18) Among the reported structural design problems solved by dynamic programming are the minimum cost of continuous coverplated highway bridge girders, single story building selection of different roof, column and foundation elements, spacing of supports of multi-span girders, thickness variations in ship plate components, reinforcing bar arrangement in continuous reinforced concrete beams and girder selection for minimum material, detailing and fabrication costs.(19-24) These examples have in common discrete variable selection and more important a sequencing of decisions into stages which satisfy the dynamic programming criterion. As an example in Figure 3, the single story optimization starts separately with the roof and then includes the column and finally the foundation and bay spacing. Since roof cost is independent of the supporting columns and foundation this sequencing of decisions is possible. The same notion of sequencing is true of the other dynamic programming examples mentioned.

4. GEOMETRY AND SHAPE OPTIMIZATION

In extending the design optimization hierarchy beyond the variables associated with design details the variables associated with shape and geometry have arisen. This is a natural continuation of much of the early optimization work on truss and frame structures which optimized the force distribution within the structure or element optimization which finds the best details in a specific cross section. Naturally much greater cost savings can be made by optimization of geometry variables than by member selection alone since the force distribution is relatively insensitive in many cases to the latter variables.

An example of this is illustrated in Figure 4 with a section of a transmission tower.(25) With a fixed initial geometry as in Figure 4a little savings can be realized in structure weight by optimizing the element force distribution. The difference between total optimized weight and say the weight obtained by a traditional direct iterative analysis and design approach is quite small even over a wide array of load conditions and even displacement constraints. However, when geometry variables are introduced the structural weight is reduced 18% as in Figure 4b. The geometry variables were the location of joints and the width of the base support. In this shape optimization the geometry design variables were separated from element design variables. For each change in geometry the minimum element design was found by a direct design method such as stress ratio. The changes in geometry were found by gradient methods.

A broader generalization of this search for optimum geometry or "best shape" structure was reported by Zienkiewicz and Campbell.(26) Starting with a finite-element analysis program an extension was programmed to automate the calculation of derivatives of structural behavior such as stresses and displacements with respect to various structural shape parameters. Optimum changes in geometry were then carried out by linear programming. Applications of this approach have been reported for arch dam geometry, dam cutouts and plates. Vitiello reported a similar program for beam shapes, gravity dams and seismic loading.(27) By properly fitting the structural behavior with polynomial functions of the geometry he was able to do the minimization with a penalty function program. An example showing the gravity design variables and the finite element modelling is given in Figure 5.

Ramakrishnan and Francavilla also using a linear programming approach found optimum shape designs for plates, pressure vessel end closures, and a gravity dam.(28)

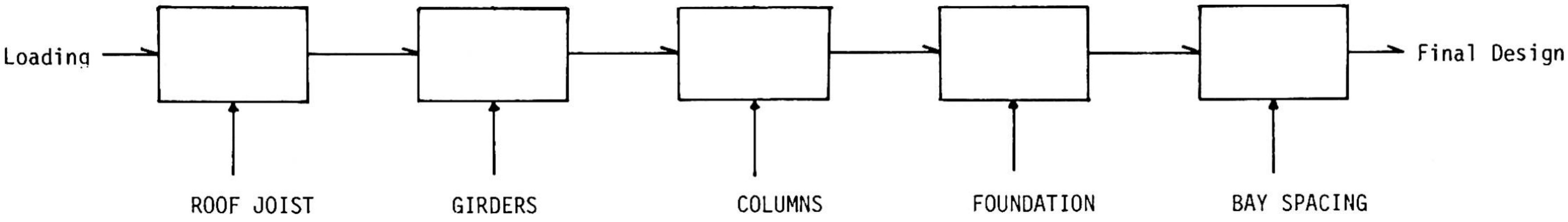
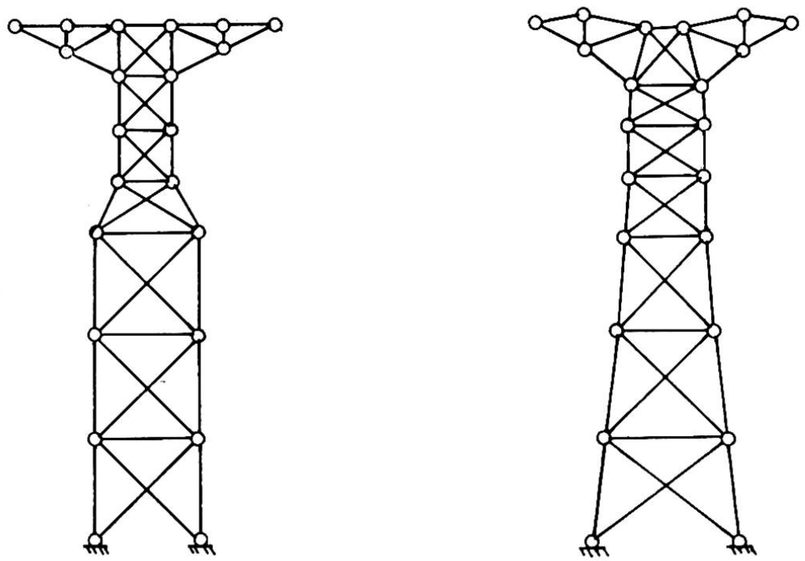


FIGURE 3: DISCRETE DESIGN OPTIMIZATION

Single story building ref. 20 and extensions by the author.



(a) Weight = 2306

(b) Weight = 1910

FIGURE 4: OPTIMAL GEOMETRY DESIGN OF TOWERS (ref. 25)

(a) Starting Design
(b) Optimal Geometry

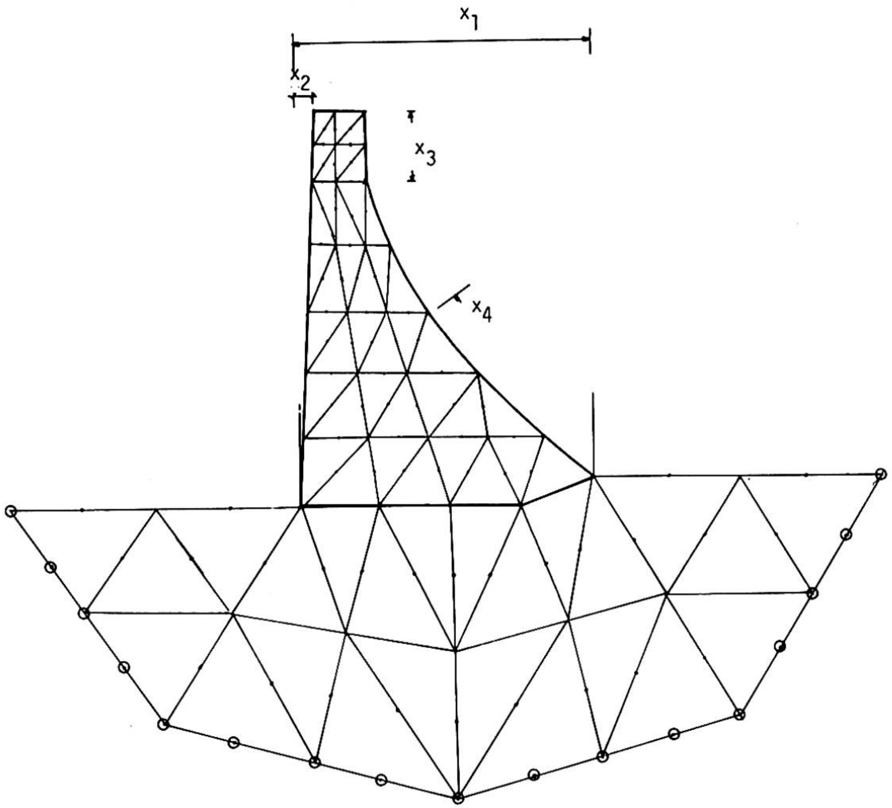


FIGURE 5: OPTIMUM SHAPE OF GRAVITY DAM,
64 Element, 262 D. of Freedom F.E. Analysis,
4 Design Variables, ref. 27.

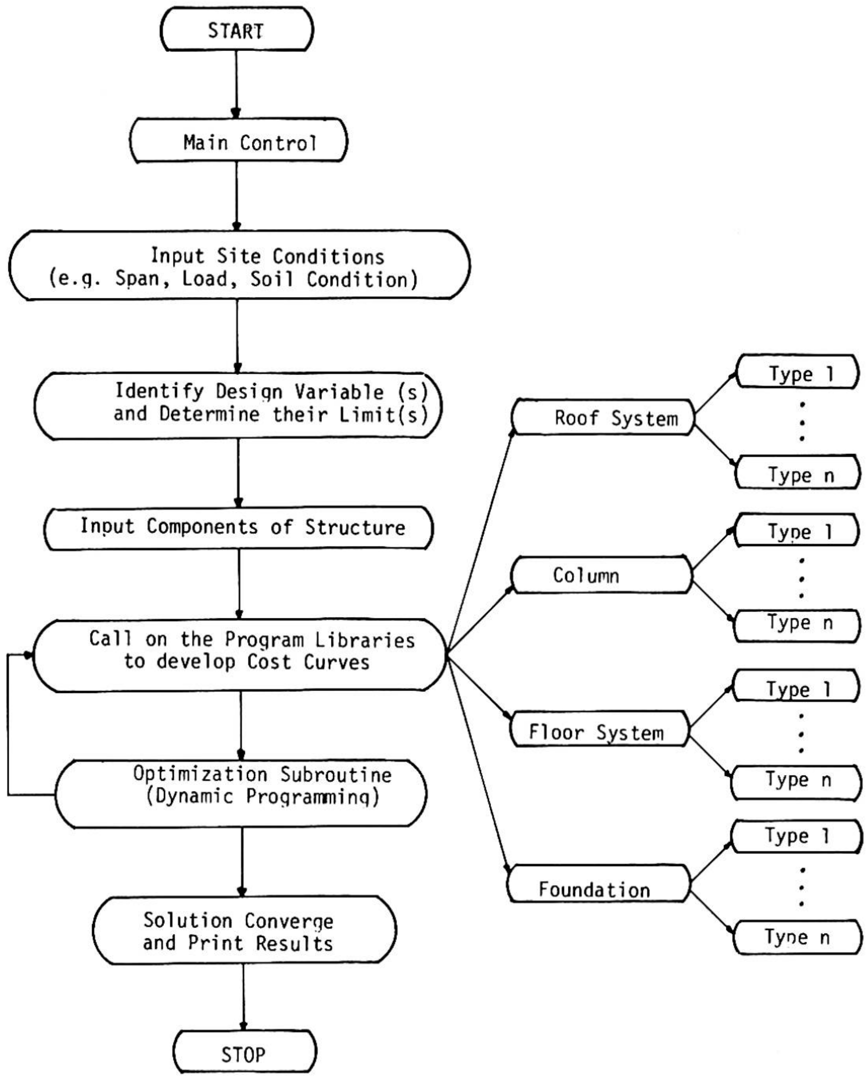


FIGURE 6: FLOW CHART FOR GENERAL SYSTEM PROGRAM FOR BUILDING OPTIMIZATION

It is apparent that despite some of this work described in shape optimization that much remains to be done and would accordingly be very profitable. Certainly, once the geometry of a structure is fixed there may be little amount of cost or material saving to be achieved by optimiztion techniques. It is therefore important that the geometry variables be considered as part of the automated design search. The most fruitful application appears to be three dimensional heavy structures such as towers, dams, pressure vessels, containment structures and storage tanks for which indeed form may follow economy and structure represents a major part of the total investment. In buildings, except perhaps for high-rise or other structures in severe seismic or climatic zones the potential material savings in the structure represents only a small percentage of the total building cost and therefore cannot dictate geometry in conflict with other architectural considerations. However, even in buildings particularly in industrial applications there may still be some gain to introducing geometry variables concerned with bay dimensions or location of shear walls and stiffening trusses.

5. SYSTEM OPTIMIZATION

The above discussion has almost exclusively centered on design variables which describe a structure but are in the main continuous variables such as element sizes or geometry variables. An exception has been the example in Figure 3 illustrating a single story building with discrete variables. What is unique about this latter example is the selection among alternative structural elements to perform the same function. Thus not only, say rolled steel beams but welded girders, reinforced concrete or prestressed beams and even trusses are sequentially compared while finding the optimum. Similarly, for other components of the structural load carrying system. Thus, the hierarchy of possible designs entering an automated optimization is considerably expanded.

Figure 6 shows a flow diagram of a general purpose program for performing the sequential comparison and optimization. It has been applied to design of single story buildings but may also be used for high-rise buildings. The various components required for the structure i.e., foundation, columns, girders, roof joints etc. are treated independently. A table is developed which obtains the minimum cost of each component as a function of the loads, with the range of possible loadings automatically established based on the input data. Each component in such a table may be found only after comparison of different types of elements to carry out the same function. These individual elements may be from a stored list of optimum designs or else generated by element design programs for a specific structure. Any convenient programming method either non-linear optimization or direct design may be used for finding the element table. The entire program is then controlled with a dynamic programming type minimum cost selection scheme to choose the best combination of elements. The input data specifies which element types should be candidates for a particular structure and new elements can be added to a program library.

To make such general programs more accessible to designers simplified programming languages are needed. The input data must be in the form of basic geometrical dimensions and load data. The connecting of components must be inputted in a direct manner as well as specifying the possible element types which are design candidates. Since the variables are so general including element selection and geometry the output need not be a detailed design specification but rather a broad general indication of the element types and geometry which give the minimum cost structure.

6. LIMITATIONS ON OPTIMIZATION APPLICATIONS

Since computers and procedures for automated optimum design have been with us for some time it may be appropriate to reflect on why such methods have not always entered routine office practice. To be sure, a similar question may be raised about matrix analysis procedures that despite readily available programs many structures are built today after being analyzed with only crude approximate techniques. A major reason, at least in the United States, is the lack of incentive for designers to utilize computer methods which cost them money but save the client construction cost. This is one reason why many of the extensive applications of computer optimization have been in areas of design-build operations where there is stiff competition and hence a desire to reduce construction cost in order to obtain a job. Other optimization applications have been to bridge construction where many bridges are designed by state highway departments who are in effect the owners of the structure.

Another limitation has been that many of the computer programs have required detailed knowledge of mathematical programming and computer software techniques. This is changing and as in more recently developed structural analysis programs, the optimization routines in some of the examples cited above do not require the user to be at all familiar with such programming procedures. In fact, in many cases, the user finds these optimum design programs easier to utilize then traditional design tools or programs. It is this attraction, rather then the construction cost saved by optimization which has often decided the user in favor of this approach.

7. CONCLUSIONS

1. Advances in automated computer techniques for design have reached the stage where many types of detailed design and selection between alternatives for minimum cost can be carried out. Such design tools could be used for lowering cost, increasing standardization of elements and evaluating effects of changing constraints.

2. Success in achieving programs for element design has suggested that the design variable search be extended to include more significant variables of material and geometry. In particular shape optimization has been used, particularly for massive concrete structures such as gravity and arch dams and containment structures.

3. System programs capable of data manipulation and automated design of a wide variety of different structural schemes can be expected during the next few years. This should make possible the application of automated design by engineers with little background in programming and software techniques. At the same time there will still be demands for special purpose programs which more efficiently automate the design of a single type of structure. This will be done by organizations which have repeated need for a particular structure and are prepared to invest time and money in computer applications. An example of this latter approach is the GAD system developed by Professor Goble at Case Western Reserve University for the design of continuous welded plate girder highway bridges.(5.19) The program has been in use by the Ohio Department of Transportation for several years. The program reflects the cost data, design details, code specification and construction practices of that organization. However, due to the number of bridges of this type which are built the investment in computer programming was justified.

4. Since the cost of developing optimization programs may be large and the incentive for the design firms to use these programs may be relatively small the advances into practice of such techniques may be thwarted. A mechanism such as a central agency is needed to develop, document and disseminate such programs to insure wide practical utilization.

8. ACKNOWLEDGEMENTS

The author would like to acknowledge the financial support given to the work by the United States National Science Foundation (GK-35412). Also, the contribution of graduate students especially J. Yeung and colleagues in this effort are appreciated.

9. REFERENCES

1. Courbon, J., Optimization of Structures, Preliminary Publication, 8th Congress, I.A.B.S.E., New York, 1968.

2. Structural Optimization Symposium - AMD Vol. 7, 1974, Editor L.A. Schmit, Jr.. American Society of Mechanical Engineers.

3. Schmit, L. A., Structural Design by Systematic Synthesis, Proc. 2nd Conf. on Electronic Computation, ASCE, 1960.

4. Goble, G. G. and Moses, F., Automated Optimum Design of Unstiffened Girder Cross Sections, AISC Engineering Journal, April 1971.

5. Goble, G. G. and Moses, F., Experience with Practical Applications of Structural Optimization, Proc. 6th Conf. on Electronic Computation ASCE, 1974.

6. Goble, G. G. and LaPay, W. S., Optimum Design of Prestressed Beams, J. Am. Concrete Inst., Vol. 68, No. 9, September 1971.

7. Joffe, I., Minimum Weight Design of Welded Gabled Frame Structures, Report No. 51, Dept. of Solid Mechanics, Struct. and Mech. Des., Case Western Reserve University, Jan. 1972.

8. Moe, J., Design of Ship Structures by Means of Nonlinear Programming Techniques, Symp. on Struct. Opt., AGARD Conf., Proceedings No. 36, 1969.

9. Stoman, Sayed, Optimization of Shear Wall Structures, Report No. 58, Dept. of Solid Mechanics, Struct. and Mech. Des., Case Western Reserve University, Cleveland, Ohio, June 1974.

10. Kirsch, U., Optimum Design of Prestressed Plates, Journ. of the Struct. Div., ASCE, Vol. 99, ST6, June 1973.

11. Moses, F., Optimization of Reinforced Concrete and Other Structural Elements. Symposium on Optimization and Automated Design of Structures. Report No. SK/M21, Div. of Ship Structures, Tech. U. of Norway, Trondheim, Jan. 1972, pp. 281-297.

12. Fox, R. L. Optimisation Methods for Engineering Design - Addison Wesley, 1971.

13. Templeman, A. B., and Winterbottom, S. C., Structural Design Applications of Geometric Programming, Second Symposium on Structural Optimization, Milan AGARD CP no. 123. April 1973.

14. Moses, F., and Onoda, S., Minimum Weight Design of Structures with Application to Elastic Grillages, Intl. Journ. for Num. Methods in Engl, Vol. 1, 311-331, 1969,

15. Fox, R. L. and Schmit, L. A., Advances in the Integrated Approach to Structural Synthesis, Journal of Spacecraft and Rockets, Vol. 3, No. 6, 1966.

16. Reinschmidt, K. F., Cornell, C. A., and Brotchie, J. F., Iterative Design and Structural Optimization, Journ. of the Struct. Div. ASCE Vol. 92, ST6, Dec. 1966.

17. Vanderplaats, G. N. and Moses, F. Structural Optimization by Methods of Feasible Directions. Computers and Structures, Vol. 3, 1973.

18. R. Bellman and S. E. Dreyfus, Applied Dynamic Programming, Princeton University Press, Princeton, N.J., 1962.

19. Goble, G. G. and DeSantis, P. V., Optimum Design of Mixed Steel Composite Girders, Journ. of Struct. Div., ASCE, Vol. ST6, Dec. 1966.

20. Aguilar, R. J. Dynamic Programming, Report of Civil Engineering Dept. Louisiana State University.

21. Aguilar, R. J., et. al, Computerized Optimization of Bridge Structures, Computers and Structures, Vol. 3, 1973.

22. Moses, F. and Tonnessen, A., Dynamic Programming for Computing Optimal Plate Dimensions in Some Ship Structures, European Shipbuilding, No. 4, 1967.

23. Hill, L. A., Automated Optimum Cost Building Design, Journ. of the Struct. Div., ASCE, Vol. 92, ST6, Dec. 1966.

24. Moses, F. and Goble, G. G., Minimum Cost Structures by Dynamic Programming, AISC Engineering Journal, July 1970.

25. Vanderplaats, G. N. and Moses, F.. Automated Optimal Geometry Design of Structures, Journ. of the Struct. Div. ASCE. Vol. 98, ST3, March 1972.

26. Zienkiewicz, O. C. and Campbell, J. S. - Shape Optimization and Sequential Linear Programming - Chp. 7, Optimum Structural Design, John Wiley (1973).

27. Vitiello, E. - Shape Optimization Using Mathematical Programming and Modelling Techniques - Second Symposium on structural optimization, Milan, AGARD CP 123, April 1973.

28. Ramakrishnan, C. V. and Francavilla, A.,Structural Shape Optimization Using Penalty Functions, Journal of Structural Mechanics, Vol. 3, No. 4.

SUMMARY

Structural optimization is defined as directed computer techniques for improving designs within well defined cost contexts and recognized constraints. Applications are divided into:
a) element design characterized by code constraints of practice;
b) system optimization involving large numbers of elements and
c) discrete decision variables.

Solutions and examples are presented for all three categories. Geometry and shape optimization as well as general programs for optimizing a variety of different structures is discussed in detail.

RESUME

L'optimisation structurale a pour but d'améliorer le dimensionnement de structures au moyen de techniques appropriées d'ordinateur, dans des limites de coûts et de contraintes bien définies. Le domaine d'utilisation en est le suivant:
a) dimensionnement d'éléments conformément aux règlements de construction
b) optimisation de systèmes composés d'un grand nombre d'éléments
c) variables discrètes de décision.

Des solutions et des exemples sont donnés pour ces trois catégories. L'optimisation de la forme et des dimensions est présentée en détail; des programmes généraux applicables à diverses structures sont également discutés.

ZUSAMMENFASSUNG

Die Optimierung von Tragwerken wird definiert als unmittelbare Anwendung der Computertechnik zum Entwurf und zur Berechnung von Konstruktionen bei genau umschriebenen Nebenbedingungen hinsichtlich Baukosten und zulässigen Spannungen. Die Anwendungsmethoden werden aufgeteilt in:
a) Bemessung von Einzelelementen nach den geltenden Normenvorschriften,
b) Optimierung ganzer Systeme bestehend aus einer grossen Anzahl von Einzelelementen,
c) diskrete Entscheidungsvariable.

Für alle drei Kategorien werden Lösungen und Beispiele angegeben. Die Optimierung der Form und der Abmessungen sowie allgemeine Programme für die Optimierung von verschiedenartigen Bauwerken werden eingehend besprochen.

Leere Seite
Blank page
Page vide

Examples of Computer-aided optimal Design of Structures

Exemples de calculs d'optimisation à l'aide de l'ordinateur

Beispiele des Computer-Einsatzes bei der Optimierung

RONALD A. GELLATLY DONALD M. DUPREE
Bell Aerospace Company
Division of Textron
Buffalo, N.Y., USA

Applied Structural Optimization

I. INTRODUCTION

The total process for the design of a sophisticated structure is a multistage procedure which ranges from consideration of overall system requirements down to the detailed design of individual components. While all levels of the design process have some greater or lesser degree of interaction with each other, the past state-of-the-art in design has demanded the assumption of a relatively loose coupling between the stages. Initial work in structural optimization has tended to maintain this stratification of design philosophy, although this state of affairs has occurred, possibly, more as a consequence of the methodology used for optimization than from any desire to perpetuate the delineations between design stages.

In recognition of this stratification, a possible hierarchy of design variable classes has been postulated.(1) The partitioning implied in this manner is not rigid but is representative of possible or probable design capabilities compared to total design requirements.

The hierarchy is

a) Member Sizes
b) Configuration
c) Material Properties
d) Construction or Topology

In the first class, all geometric details of the structure are fully defined and only member sizes are to be chosen by a design process. Although apparently a very restricted class of problems, this actually represents (i) the limit of most of the optimization capability available to date and (ii) an extremely wide class of structural problems. It is a fact that in many structures the location and configuration of a great deal of the primary structure is mandated by nonstructural considerations. Likewise materials and construction will frequently be dictated by environment, design codes, cost, etc. There are many other structures for which the above does not apply.

By configurational variables, location but not number of principal components is implied. Hence, the first two members of the hierarchy may be regarded as continuous variables*, whereas the latter two categories clearly involve discrete noncontinuous variation of parameters. Because of the difficulties encountered in dealing with noncontinuous variables within a mathematical framework, principal attention has been generally confined to the first two classes of variables, with maximum attention on the member sizes.

The principal approaches to the optimization of structural systems for minimum weight in the past have been based upon the use of a combination of mathematical programming or other rigorous numerical search techniques and an equally rigorous structural analysis method. There have been many variations on this theme, but the essential combination of methods has remained the same. For analysis, finite element methods have been the most frequent choice, while the numerical search techniques have run the gamut from linear programming to Monte Carlo.(2,3,4) While this type of combination of methods is valid and appropriate for certain classes of problems, within the individual strata of the overall design process, it has led to certain intractable situations.

The rigor and sophistication of both the analysis and search procedures inevitably mandate numerical complexity and large computer costs for the optimization of anything approaching a representative large scale system. This, in turn, has cast considerable doubt upon the economic value of some optimization concepts. While many difficulties have been encountered using traditional methods of mathematical programming, there have been significant developments in new approaches to structural optimization which have overcome some of these difficulties for selected classes of problems.(5,6,7,8)

While mathematical programming methods are fairly rigorous and extremely general in their range of applicability, computer programs developed along these lines tend to be effective for the optimization of small scale systems only. When expanded for the optimization of realistic large scale structures, such approaches tend to become excessively costly and also of doubtful reliability and accuracy. The major problems seem to arise from a large increase in the number of analysis iterations with increase in the number of design variables. In addition, the explicit or implicit need to calculate numerical approximations to derivatives of constraints with respect to all variables means that each iterative step itself becomes lengthy.

Some new developments in mathematical programming have tended to overcome some of the difficulties but others remain. In addition, the possibility of further new developments in both analysis and numerical search techniques cannot be overlooked.

*It is recognized that in many branches of structural engineering, principal members may only be selected from standard sizes and are not strictly continuous variables. This problem is usually treated by considering section properties as continuous variables and then selecting the nearest standard sizes for the final designs.

One approach to the problem which apparently avoids many of the pitfalls of mathematical programming is through the use of optimality criteria formulations.[5,6]

The basic concept behind otimality criteria is the rejection of the generality of mathematical programming and the utilization of the physical characteristics of the structural optimization problem to generate an approach of somewhat limited applicability but of the greatest computational efficiency.

In the optimality criteria approach, preconditions regarding the optimum structural system are generated based upon a physical, mathematical or even intuitive understanding of the problem. A simple search procedure is then developed to find the design satisfying these specified criteria.

A full discussion of this approach to one facet of structural optimization is given in Section II along with examples of the applicability.

Even with the development of optimality criteria programs and other similar approaches, these methods still suffer in many cases from severe limitations with regard to class and ranges of design parameters which can be treated as variables in a search for an optimum system.

It is this latter fact which has tended to maintain the stratification of the design process. It has been simply not possible or practical to mix variables of the different hierarchy classes in any rigorous search procedure. The major handicap has been the lack of continuity of variation of some parameters. While the concept of fixing configuration, mode of construction and materials at the outset of design may be acceptable for some structures, it will certainly fall far short of a goal of overall system optimization. Attempts have been made, with varying degrees of success, to incorporate configurational variables.[2,3] Generally, the stumbling block to the use of configurational and other variables (apart from computational costs) has been the requirement for continuity of variation in the parameters, due to the need for derivatives to provide search directions in a continuum space. With configurational variables this may be marginally possible provided the topology is undisturbed but to effect continuous variation in such concepts as material properties, construction mode and topology is beyond the capabilities of the vast majority of mathematical programming techniques.

For the optimization of large scale systems where many or all of the above parameters are initially undefined, more flexible and more general approaches have been sought. An additional consideration has been to develop an approach which would avoid the high computational costs of the more rigorous formulations, providing thereby an economic tool for ready use in design trade-off studies.

One new approach to the determination of the minimum weight of complex structural systems involving material, constructional and configurational variables in addition to the more conventional design variables has been developed and is labelled the "sieve-search" technique.[9] In this new procedure, which sacrifices some degree of rigor for economy and generality of application, an attempt has been made to consider the effects of detailed design on the overall configuration of the total system and thus tie together hitherto uncoupled design stages.

In performance of optimization studies using the sieve-search technique, the guiding philosophy is the generation of an optimal arrangement of pre-optimized components. In this approach, the detail components of a structure are optimized first using local loading conditions and then the major configurational parameters are varied in order to find the optimal arrangement of the locally optimized components. The optimal design is obtained by a sequential comparison of the individual designs based on discrete values of configurational and constructional design variables. The above procedure is labeled a sieve-search since all nonoptimum designs are eliminated by the sequential comparisons leaving only the least weight design. The process can be labeled "discrete" in contrast to the more classic approaches wherein continuous variables are treated.

The sieve-search method was developed initially for an applied to the design of an extensive class of surface effect ships. Section III discusses the basic philosophy behind this approach to structural optimization using the surface effect ship as a prime example. The extension of the procedure to other classes of structural design problems is both possible and economically attractive. Its potential use for bridge design is also discussed in Section III.

II. OPTIMIZATION USING OPTIMALITY-CRITERIA

As discussed previously, there are a number of basically different approaches to the problem of overall structural optimization. While some of the variations in the approaches stem from differences in the classes and types of systems which are being optimized, there are also problems for which two or more methods of solution are available.

A classic problem, of great practical interest, is the optimization of a structural system whose overall geometry is fully defined and fixed by a set of external conditions but whose member sizes are to be selected optimally. The structure will usually be subjected to a multiplicity of loading conditions (no one of which is uniquely critical) and in addition to known limitations on the strengths of individual components, stiffness of the system may be of critical importance. Also fabricational constraints or other codes may mandate minimum sizes for constituent members.* For this type of problem which is encountered frequently in engineering design, the primary approaches to optimization developed during the 1960's were based upon the use of mathematical programming

*In discussing a structure, the concept of an assemblage of individual elements is used. This is generally consistent with the idea of a finite element model which is usually used for the actual structural analysis. If a continuun is considered, it, too, would be represented as an assemblage of discrete elements,which may be viewed as separate variables in an optimization process.

formulations.[4] Because of the computational difficulties encountered with mathematical programming, this approach has been largely abandoned for large scale structures and newer methods based on the concepts of optimality criteria have been developed.

The underlying concepts behind optimality criteria methods can best be illustrated by considering the contrasts between optimality criteria and mathematical programming. In mathematical programming approaches, sets of rules are established for numerical search procedures which will determine an optimal solution in a strictly empirical manner. The set of search rules will guarantee a continuous and monotonic decrease in a prescribed merit function, essentially without regard to the physical (or sometimes even mathematical) nature of that merit function. The search will regard constraints, if such exist, and will continue searching until no further improvement in the merit function is possible. No preconditions concerning the nature of the optimum are specified beyond the criterion that it is impossible or uneconomic to determine a further design which will be an improvement on the present design. This approach may be labelled post hoc since the optimum is identified essentially only on an after-the-fact basis. Both the strength and weakness of mathematical programming reside in this concept. The strength is the generality which this independence of mathematical formulation imparts, with the resulting wide range of applicability; the weakness is that no use is made of any of the physical characteristics of the problem and hence frequently an unnecessarily long and costly solution process results. The antithesis of this situation arises in an approach which recognizes the physical nature of the structural optimization problem per se and sets out to take fullest advantage of the restricted class of problem. In this approach, some conditions are established initially concerning the nature of the design which will be regarded as optimal. These conditions, which are defined before initiation of the redesign process, may be rigorously exact, approximate or even intuitively assumed. The essential requirement is that their application will lead to a relatively simple (usually iterative) algorithm for a redesign process converging on the design which satisfies the initially prescribed criteria. This approach is then labelled a priori, since the characteristics of the optimal system are specified initially.

The classic and most obvious example of an optimality criteria approach is the time-honored fully-stressed-design. Every practical engineer is fully aware of and would probably support the basic idea that the most efficient (optimal) design is one in which every member is used to its fullest extent under at least one loading condition. Prior to the advent of computers and the development of advanced methods of structural analysis using finite elements, generations of structural engineers have traditionally attempted to inject some degree of optimality into designs by analyzing a trial structure, using some appropriate quasi-classical procedure, and adjusting member sizes to eliminate over- or under-stressing. The more ambitious engineers might even have re-analyzed and re-sized the structure one or more times. Probably very few practicing engineers ever wondered whether anything is invalid with this rather natural 'calibrated-eyeball' approach. Accurate analysis of indeterminate structures presented a difficult problem prior to the introduction of computers, discouraging the repeated use of more elaborate schemes, while approximate analyses were somewhat insensitive to the crucial effects of rerouting internal force distributions resulting from resizing iterations.

With the appearance of computers in the fifties, the first attempt at automated optimum sizing was the computerized version of the above procedure, initially still relying on time-honored approximate analysis methods. The ensuing development of the finite element methods by the early sixties made rather accurate analyses possible for indeterminate structures of virtually any form or shape. Instead of just two or three resizing cycles now a much larger number of cycles became feasible, at least for numerical experimentation by researchers, even if not in practice.

This simple and intuitive concept was eventually formalized as the fully-stressed-design (f.s.d.). To achieve f.s.d., the most commonly used algorithm, although not the only one available[10], is the simple stress-ratio. In the stress-ratio algorithm, it is assumed that the gross forces in any member of the structure will not vary with member size and hence the member properties may be adjusted directly in the ratio of the actual to the allowable stress. In indeterminate structures, changing member properties generally effects some redistribution of internal forces, so that an iterative process is required to achieve a f.s.d. The most important feature of the stress-ratio, and other similar algorithms is, that, in marked contrast with direct numerical search procedures, the number of re-analyses needed to reach an apparently converged design is usually small and independent of the size of the problem. This intuitive approach fulfilled a need for automated sizing for strength requirements and the strength optimization problem seemed to be under control.[11,12] No such simple and efficient method existed at that time for stiffness related problems.

In the late fifties, nonlinear programming methods were introduced as the correct framework for the general structural optimization problem.[13,14] With the development of these more rigorous methods, which were applicable to both strength and stiffness constraints, it was shown that f.s.d. is not necessarily the correct optimal solution for indeterminate structures. On the other hand, it was also shown that f.s.d. may indeed frequently be a correct solution, or more importantly from an engineering viewpoint, may be a close approximation to the correct solution. Thus with f.s.d. a very efficient but invalid method of strength optimization is provided. Fortunately not too many practicing engineers are inclined to question the rigor and validity of f.s.d. and merely welcome its advantages.

The standard f.s.d. stress-ratio redesign algorithm tends to drive a structure towards a design with the stiffest routing of internal force flow, which may or may not coincide with the optimal force flows. This trend may not become apparent if only a few resizing cycles are performed and because they do usually tend to produce a succession of improved designs, they are of great value to the engineer.

The potential sources of problems with f.s.d. are quite easy to point out, but the extent to which they are present in any given situation is extremely difficult to assess. The difficulties can be demonstrated in two small example research problems, where comparison with correct solutions, obtained by numerical search, is possible. The two examples may be regarded as somewhat pathological but even for these problems it is not entirely clear what the true nature of the pathology is. Hence, it is not possible to state categorically that any real system does not contain the same disturbing influences. In the stress-ratio algorithm, only the constraints (stresses) themselves are considered and no reference is made to the factors of relevance to the merit condition, such as density. Thus f.s.d. is completely insensitive to favoring structural elements according to their strength to weight ratios. Therefore, f.s.d. tends to break down in structures which contain materials of different densities or markedly different allowable stresses. The first example (Figure 1) is of two parallel bars sharing a single load. One bar is of steel, the other is of aluminum but both have the same allowable stress. The stress-ratio algorithm will increase the size of members with higher material stiffness and/or lower allowable stress. In this example the aluminum bar will vanish and the steel bar will be retained. Clearly this is a f.s.d. but not a minimum weight design. If both bars are made of the same material, but with different allowables, the algorithm will eliminate the higher strength bar, again a poor design. It should be noted that the optimal solution for these two problems is the other bar fully stressed. The difficulty here lies with the stress-ratio algorithm, rather than the concept of f.s.d.

A second more elaborate example is the 10-bar truss shown in Figure 2.[15] The truss has a single loading case and initially the stress limit in all members is ±25000 psi. The f.s.d. obtained using stress-ratio weighs 1593 lb which is known to be optimal. Successively raising the allowable stress in bar No. 10 to ±30000 psi, ±50000 psi and ±70000 psi and again using a stress-ratio, designs of 1545 lb, 1725 lb and 1725 lb, respectively, are generated. The 1545 lb design is also known to be optimal but the last two solutions of 1725 lb are clearly unreasonable and considerably in error. In these two cases stress-ratio has tried to eliminate the high strength bar, resulting in the poor designs. Using mathematical programming techniques[8,16], the optimal design for the two high strength (50000 and 70000 psi) cases is known to be 1497 lb. Further examination of the problem reveals the interesting fact that, in both the stress-ratio (1725 lb) and the optimal (1497 lb) designs, all members are either at their full allowable strengths, or at their minimum sizes, except for bar No. 10. In each case the stress in bar No. 10 is 37,5000 psi, although both designs are radically different. Assigning an allowable stress of ±37,500 psi to bar No. 10 and again applying stress-ratio, results in a third fully-stressed-design, weighing 1568 lb, which is quite different from the other two. Clearly the whole field of f.s.d. needs further research. Some studies have been conducted and variations on the stress-ratio algorithm have been proposed[17], but with limited success.

On the other side of the coin, large scale programs have been developed, basically using f.s.d. and these programs have been successfully applied to the design of real structures. Whether or not such structures are truly optimal is somewhat academic when it is realized that such designs obtained at moderate computer cost are undoubtedly superior to those generated by hand.

Figure 3 presents the computer-generated plot of the finite element idealization of a complex wing structure. The total model had 5397 finite elements, 4104 displacement degrees of freedom and 20 separate loading conditions. Redesign studies were performed on the inboard half of this structure starting from various initial designs. The model considered had 3275 finite elements (design variables) and 2520 displacement degrees of freedom. Only two loading conditions were considered critical for sizing. In all cases only three iterations were performed showing acceptable convergence. Six iterations would have been sufficient for accurate production work. The program used for this optimization was ASOP[18] and for the three iterations required 6000 seconds CPU time on a CDC6600 computer.

The preceding discussion has dealt rather extensively with f.s.d. because this is the classic example of an optimality criterion, and it is an approach to optimization which is widely recognized and accepted. It is, nevertheless, very limited in its use. Its role as an optimality criterion, per se, would probably not have been recognized, if there had not been a pressing need for the development of suitable and efficient optimization procedures for stiffness constraints. The driving motivation for the exploration of optimality criteria methods for stiffness constraints was the excessive cost of using direct numerical search methods. What was sought was an approach as simple as stress-ratio but for displacement constraints. Optimality criteria were investigated since such concepts, by definition, contain gradient related information as a result of their derivation. By taking full advantage of the special structural properties of the problem, these criteria should lead rapidly and efficiently to the solution.

The actual development of a practical method for stiffness constraints was a multistage process in which many researchers individually contributed key concepts[19,20,21,5]. It is not of relevance here to discuss all the stages in this development progression; a fuller description may be found in Reference 22.

The essential step in the development of the currently used approach to stiffness constraints was the formulation of a single displacement constraint problem using a Lagrangian multiplier. In a structural system with fixed geometry, A_i, the characteristic sizes of constituent members, are considered to be design variables. If $W(A_i)$ is the merit function for the structure and $F(A_i)$ is a single displacement which is to be constrained to have a magnitude C, then values of A_i which minimize W, while satisfying the equality constraint can be determined by use of a Lagrangian multiplier formulation. The expression

$$W^* = W(A_i) - \lambda \left[F(A_i) - C \right] \qquad (1)$$

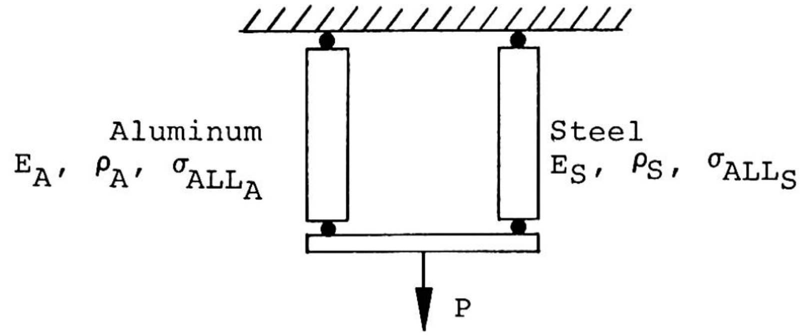


Figure 1. Parallel Bars of Aluminum and Steel

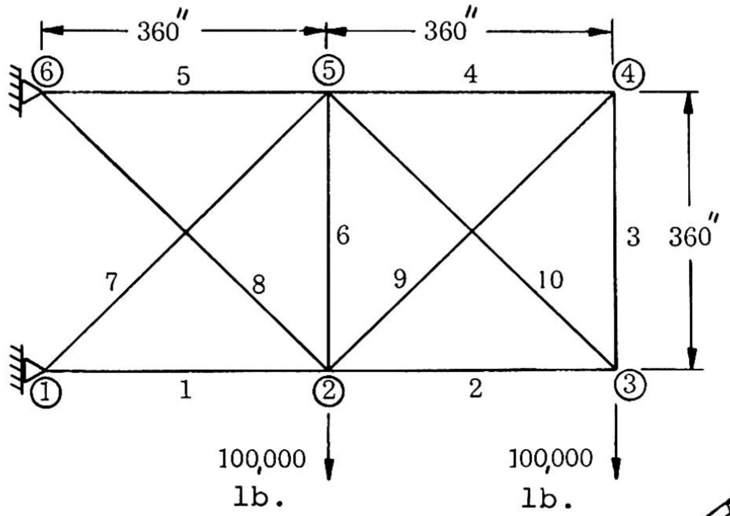


Figure 2. 10 Bar Truss

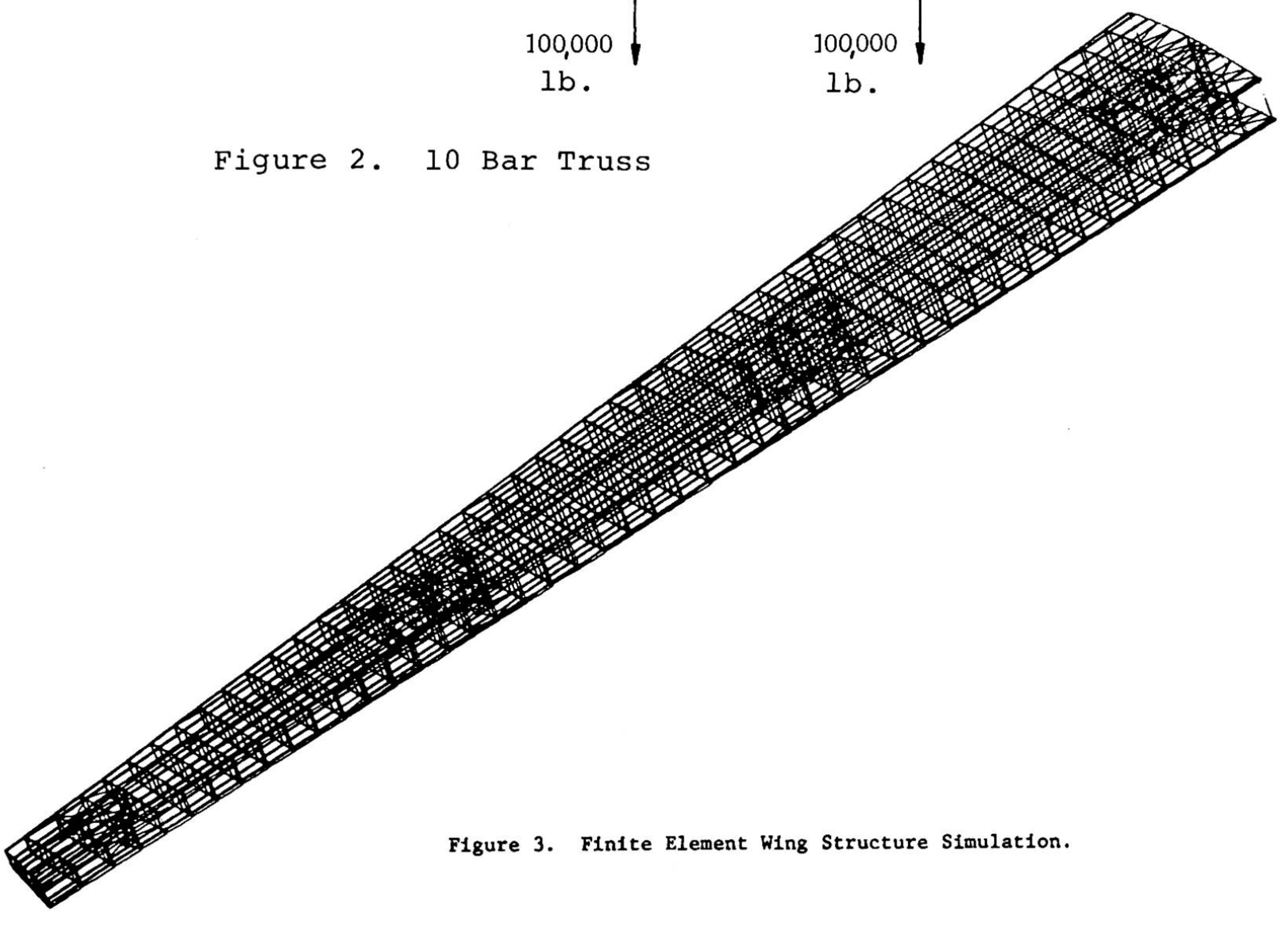


Figure 3. Finite Element Wing Structure Simulation.

is written and differentiated with respect to A_i to yield

$$\frac{\partial W^*}{\partial A_i} = \frac{\partial W}{\partial A_i} - \lambda \frac{\partial F}{\partial A_i} = 0 \qquad (2)$$

Equation (2) is then the necessary condition for the optimum system, or the optimality criterion. For specific classes of problems, it can be proven that this condition is also sufficient for global or local optima.

Equation (2) can be rewritten in the more revealing form

$$\frac{\partial W/\ \partial A_i}{\partial F/\ \partial A_i} = \lambda = \text{constant, for all } i \qquad (3)$$

Written in this form, there can be seen the valuable and relevant information that, in an optimal structure, the change in the measure of the behavior (displacement) for a unit change in the measure of merit is the same for every free variable. That is, the cost of improvement in the design is the same for every member in the optimal system. This statement is quite general and applies to the optimization of a structure for any type of merit function (weight, cost, etc.) and for any type of constraint which is characteristic of the structure as a whole. Thus, not only displacement constraints can be considered, but also overall buckling, dynamic response, flutter and any other phenomena which are indicative of total structural response.

By the same token, strength constraints do not satisfy the criterion of Equation (3), since they are, of necessity, individual characteristics of the constituent members and not of the structure in toto.

In order to translate Equation (3) into a working procedure for the stiffness optimization of a structure represented by an assemblage of finite elements, some particularization of the general definitions used previously is necessary. It is assumed that both merit and stiffness of the system are linear functions of the design variables A_i. These specializations are not necessary; they are made only to simplify the expressions for the cost and constraint function derivatives for a concise presentation. Other functional relationships are possible. One additional specification is crucial to the derivation of the final simple numerical procedure. This requirement, which is generally satisfied by most analytical methods, is that both the total cost and total stiffness be sums of individual members contributions. As a result, the simultaneous equations implied by Equation (3) uncouple for each value of i and can hence be solved in an extremely expedient manner using simple recursion formulae. The not very widely recognized importance of these key considerations, satisfied fortuitously by finite element analysis techniques, is that they remove obstacles which hitherto existed to the use of classical Lagrangian multiplier formulations for structural optimization. It is assumed in the following brief development that the complete behavior of the structure is analyzed using the finite element displacement method. In accordance with the above definitions, a merit function (weight) is written

$$W(A_i) = \Sigma w_i = \Sigma \bar{w}_i A_i \tag{4}$$

Similarly the stiffness behavior is written

$$F(A_i) = \Sigma e_i = \Sigma \bar{e}_{i/A_i} \tag{5}$$

Equations (4) & (5) merely express the linear summations discussed previously. $\bar{w}_i$ & $\bar{e}_i$ are the contributions of individual unit-sized elements to the total weight and stiffness of the system. For a simple bar element with A_i as the cross-sectional area

$$\bar{w}_i = L_i \rho_i \tag{6}$$

where L_i is the bar length

and ρ_i is unit material cost (density).

For other types of elements A_i & L_i must be appropriately defined, but the general form of Eq. (6) still holds.

The stiffness of a structure under an actual loading system (P) is computed by imposing a virtual unit load system (Q) in the direction of displacement required, and computing the virtual work of system. The contribution of each element is given by

$$e_i = \delta_i^{Pt} K_i \delta_i^{Q} \tag{7}$$

where $\delta_i^{P,Q}$ are the vectors of the nodal displacement of i^{th} element due to the actual and virtual loading systems, $K_i = k_i A_i$ is the stiffness matrix of the element and k_i is the unitized element stiffness matrix.

For other types of stiffness related constraints, such as buckling, vibrational response, etc., corresponding relationships to Eq. (7) can be derived and used in the subsequent development of a suitable redesign algorithm. Examples of buckling and dynamic response constraint formulations can be found in References 23 and 24. Substituting the above relationships into Eq. (2) and after some algebraic manipulations the recursion relationship is obtained

$$A_i^{\nu+1} = \frac{A_i^{\nu}}{C^*} \sqrt{\frac{\delta_i^{Pt} k_i \delta_i^{Q}}{L_i \rho_i}} \qquad \sum_j A_j^{\nu} L_j \rho_j \sqrt{\frac{\delta_j^{Pt} k_j \delta_j^{Q}}{L_j \rho_j}} \tag{8}$$

where the superscripts ν , $\nu+1$ indicate the values of A_i at successive iterations and C* is the prescribed value of the stiffness.

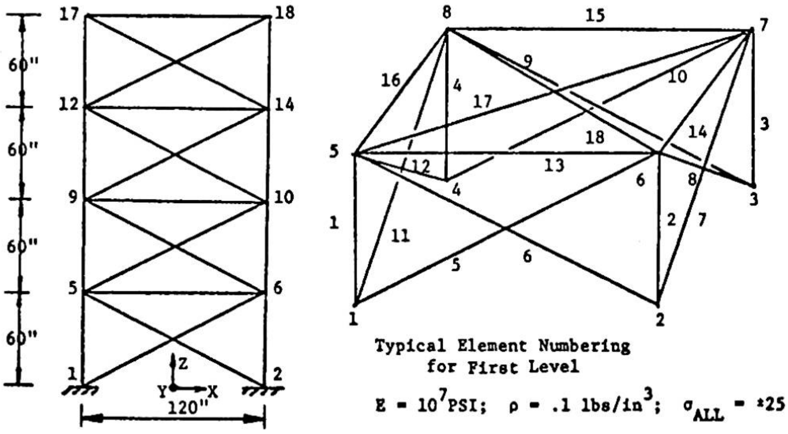
Eq. (8) is then the redesign algorithm for a single stiffness equality constraint. In order to generalize the algorithm for multiple inequality constraints, the recursion relationship is applied to each constraint in turn and then the dominant values of A_i are selected for each member. The redesign process is iterative at each stage and a procedure for partitioning design variables into active and passive groups is used to select which members are effectively design by which constraints. This algorithm, known as the envelope method, also permits the simultaneous consideration of strength and minimum member sizes. The envelope method is an obvious simplifying approximation and does not strictly satisfy the correct optimality criteria for multiple constraints. It basically disregards the sizing given by one constraint when satisfying another.

Thus, analogous to the case of f.s.d., a procedure has been obtained for stiffness redesign based on an approximate criterion which has the merit of great simplicity and good general behavior. Experience has shown that the solutions for stiffness constrained problems obtained using the envelope method usually compare very favorably with more rigorous solutions obtained otherwise at much greater computational cost. The convergence characteristics of the envelope method are similar to those of f.s.d. with usually rapid convergence in a very small number of iterations, apparently independent of problem size.

A number of computer programs using optimality criteria algorithms have been developed. The program OPTIM II[6] is a large scale program which contains eight different finite elements in its basic library and is capable of application to a considerable variety of large scale problems. The elements include bars, beams and plates of various types. The program also contains a number of special features such as provision for linking elements, plate buckling computations and other capabilities intended to simplify the analyst's work.

The capabilities of such optimization programs can be best illustrated by a few example problems. These problems are generally small scale, but are intended to demonstrate the potentialities of the programs rather than to overwhelm by sheer size of problem alone. The programs themselves are only really limited by available computer size and the price (in terms of numbers of analyses) which the designer is prepared to pay.

The first example (Fig. 3) is of a simple four-level tower structure, composed of 72 primary members. The tower is subjected to two loading conditions as indicated. For obvious reasons it is desired to maintain the double symmetry of the structure, although the loading itself is nonsymmetric. The automatic linking feature is used to tie together elements where necessary. There are stiffness constraints to ensure that the tower does not sway too much under load. Figure 4 indicates that only four analyses were required for convergence. The efficient redesign logic at each stage requires only 10-15% of the analysis time.

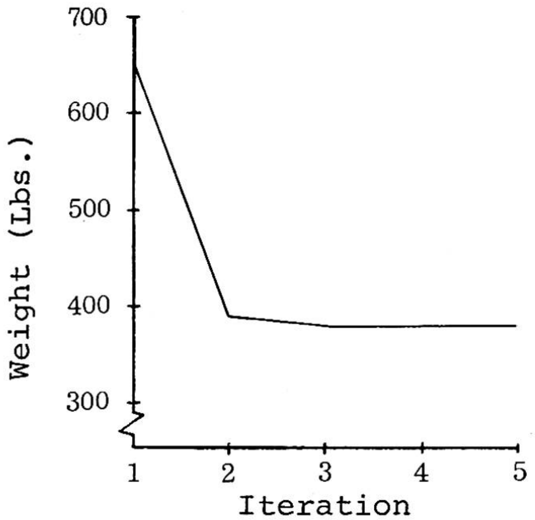


$E = 10^7 PSI$; $\rho = .1$ lbs/in^3; $\sigma_{ALL} = \pm 25$ KSI; $A_{MIN} = .1$ in^2

Displacement Limit = .25 in (All Displacements)

LOAD CONDITION	NODE	DIRECTION X	Y	Z
1	17	5,000	5,000	-5,000
	17	0	0	-5,000
	18	0	0	-5,000
2	19	0	0	-5,000
	20	0	0	-5,000

(a) Geometry and Loading



(b) Iteration History

Figure 4. 72 Bar Four Level Tower

The second example is the geodesic dome of Figure 5, designed for both strength and stiffness constraints. In this problem involving 156 elements, the dome was subject to a uniform vertical load and the vertical displacement of the central point was limited. This problem was studied using various optimization programs available and full details of the results may be found in Reference 22.

Figure 6 represents the idealization of a wing carry-through structure on a large heavy swing-wing aircraft. The loading arises from operation with the wing in two different positions. The loadings on the pivot points were then principally flexural for the wing in a forward, unswept position and torsional with the wing fully swept back. In order to maintain the aerodynamic characteristics of the wing, the rigidity of this structure must be very high. Severe limits are therefore placed on the allowable displacements and rotations of the pivot points. Initially a strength only optimization was performed yielding a weight of 5035 lb in 50 iterations. This is a very slow convergence but it should be noted that a weight of 5049 lb (0.3% heavier) was reached by iteration 18. The structure was then reoptimized with both strength and displacement constraints. The least weight of 6159 lb was reached at 50 iterations, with the same slow convergence, but 6216 lb (1% heavier) was obtained at iteration 14.

If all members of the initially obtained strength-limited design had been directly scaled to reduce the displacements of that design to meet the specified stiffness constraints, the structure would have weighed 7961 lb, over 29% heavier than the actually optimized structure. This indicates the redistribution of material effected by the optimization algorithm.

In this example, a bar idealization has been used for simplicity, but in the actual structure, plates and shear webs would be used. This raises an important point in structural optimization regarding the influence of the idealization on the optimal system. All redesign logic, for both stress and stiffness constraints is eventually predicated upon the detailed internal stresses in the individual elements. Finite elements, or indeed any other numerical analysis techniques, by their very nature introduce a certain degree of approximation into a solution. Finite elements are a piecewise representation of a continuum and certain approximating assumptions are essential to their basic derivation. The actual errors introduced into a given analysis using finite elements is usually very small and hence the results obtained are perfectly satisfactory for an engineering analysis. The widespread use and acceptance of finite element methods is a testimony to their validity.

For optimization, where many analyses may be performed and each redesign is dependent upon an erroneous analysis, the effect of the inaccuracies may be cumulative. This does not imply that the final system will be unsafe, but merely that the optimization of a structure modelled by two slightly different idealizations could result in two radically different designs. Care must be exercised in the development of optimization programs to ensure that only the most accurate analysis techniques are used. In finite element analyses, bar elements are exact and involve no

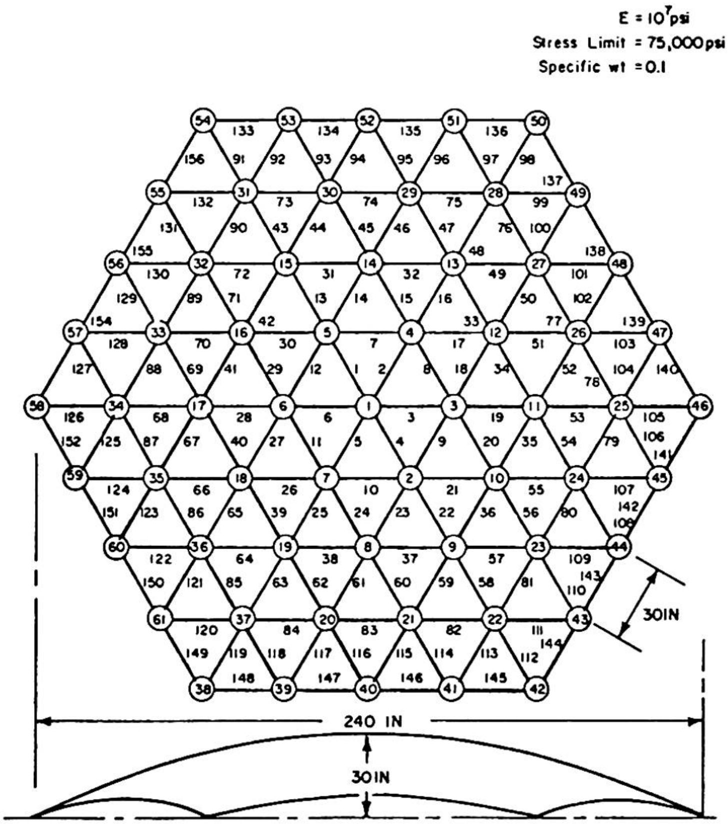


Figure 5. Geodesic Dome

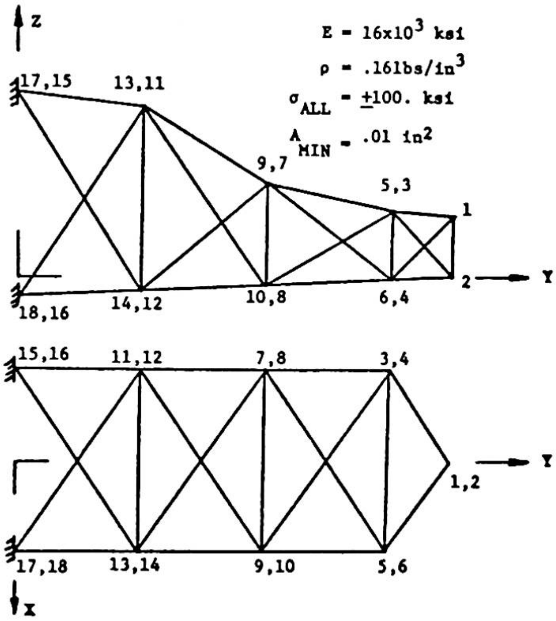


Figure 6. 63 Bar Wing Carry-Through Box

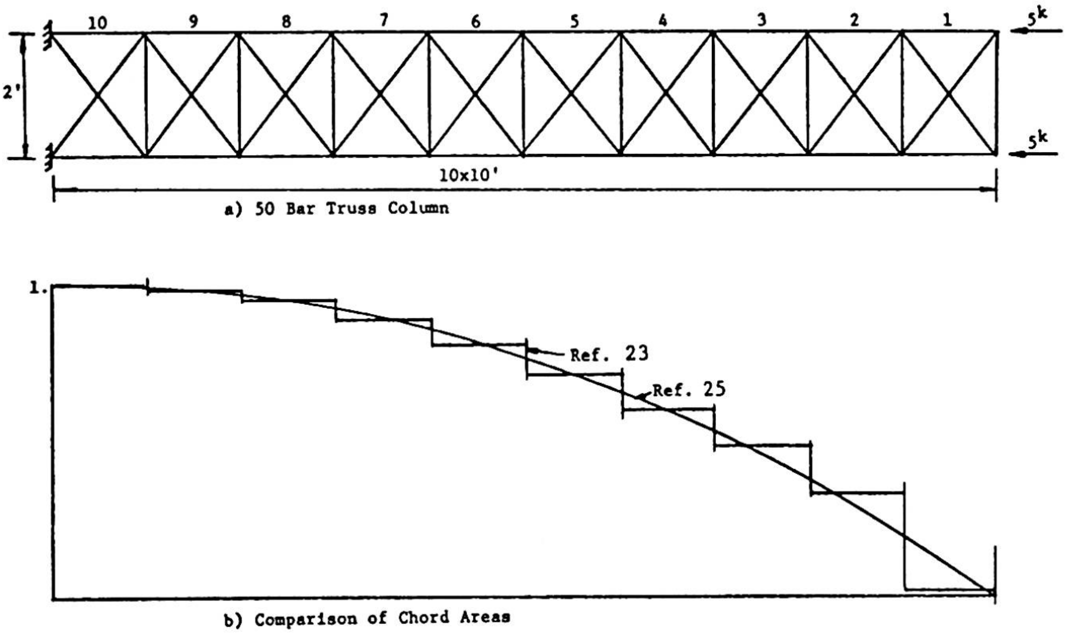


Figure 7 50 Bar Truss Column Problem

approximations. They are therefore frequently used for demonstration problems since they invite direct comparison of optimization solutions obtained by other methods by eliminating idealization errors.

The final example presented is that of the buckling of a simple laced column (Figure 7). The column has 50 bar elements and was optimized using a stiffness representation of the eigenvalue buckling problem.[23] The areas obtained for the chord (axial) members are shown plotted in Figure 7 in comparison with the exact solution obtained for the face sheets of a similar sandwich column.[25] The comparison is very encouraging.

III. OPTIMIZATION USING SIEVE-SEARCH

The selection of a truly optimal design to satisfy a particular set of engineering requirements is a complex process which strictly involves the consideration of all the classes of variables discussed in Section I. The approach presented in Section II deals with a more limited design problem in which geometry, material and construction are assumed to have been fully defined. A major question indeed must be on what basis will these governing design characteristics have been selected.

While it is true, that many psuedo-design parameters such as materials and construction cannot be treated as continuous variables and hence cannot be incorporated into any standard mathematical programming search technique, other considerations do enter into the picture. For the vast majority of engineering systems, only a limited number of materials really come into consideration. For civil engineering primary structures, titanium or boron-reinforced plastics, for example would have little or no applicability. Similarly reinforced concrete is seldom to be found in aerospace structures. Thus although there may be a potentially large number of possible materials and construction types, engineering practice and experience will indeed limit these to a finite set, which may be considered discretely. In a similar manner, although some aspects of the structural configuration, as defined by the arrangement and location of the principal structural members, are parameters to be selected by the designer, certain configurational characteristics will be absolutely defined by the service requirements of the structure. In addition, aesthetics and engineering codes will probably place some restrictions on other variables. The net result again is the specification of a finite set of configurational parameters. Finally the detailed design of individual structural components is governed by the critical loading which they experience locally. This critical loading may either arise from overall structural loading or may be a purely localized loading system which has little influence on the structure as a whole. Thus the optimum design can be generated for a given component under a specific loading system in isolation. Extending this concept, a range of optimal members can be predesigned in some suitable manner for appropriate ranges of applied loadings and sizes. This then is a so-called data bank. An example of a data bank is a structural handbook, which specifies appropriate code sizes of beams, columns, etc., for given applied loadings. It is well recognized that internal loading distributions are not strongly influenced by small variations in member properties. Hence, except for highly pathological problems of the type discussed in the previous section, it is assumed that only a very limited number of redesign iterations is required for a satisfactory degree of convergence. If the critical loading is purely local, convergence is achieved almost immediately.

With the above considerations as guides, an approximate optimization procedure for large structures was developed.[9] The guiding philosophy in this sieve-search approach is that the optimum system is an optimal arrangement of pre-optimized components.

Individual components are optimized initially under local loading conditions and the potential designs stored in a data bank. A program is then set up which cycles sequentially through all the finite combinations of the major variables. For each configuration so defined, or segment thereof, an optimum design is generated using the data banks and compared with the best design available at that point. The best design is retained and the cycling is continued.

The efficiency of this process is then highly dependent upon the data banks available. These banks contain properties of optimized components generated either by classical methods of optimization or selected from standard structural codes. An additional, but nonetheless important facet of the preset technique is the use of simplified engineering analysis methods where-ever possible during the iterative phases of the redesign cycles. Herein lies the efficiency of the sieve-search technique whereby literally hundreds of redesigns are rapidly made for selected configurational variables from which the optimum is obtained.

As a prime example of the sieve-search technique its application to the design of class of surface effect vehicles (SEV) is considered initially. The extension of the procedure to other structural systems is discussed later with particular emphasis on bridge structures.

Figure 8 is an actual photograph of a surface effect vehicle which is prototypical of an extensive class of high speed cargo vessels. Although operating in a marine environment, SEV are essentially aircraft-type structures which must be supported on a cushion of air. The development of least weight structures is therefore of prime importance in the design of such vehicles since the economic viability of SEV are dependent on low structural weight.

Before initiating the design process consideration must be given to the classes of parameters which would realistically be regarded as variables in performing the actual design. Thus external envelopes would be fixed by hydrodynamic and performance requirements - although some trade-off studies between configurations and performance might be desired. Figure 9 indicates the general form of the external craft envelope.

Constructional materials and modes may be fixed or may be selectable from a limited class of candidates. Environmental considerations will narrow the number of available materials and for each material only a very small number of constructional modes is technically feasible.

The internal arrangement of longitudinal and transverse beams and bulkheads will have been fixed in an overall sense, but the individual spacings and sizes will be treatable as free variables. The only possible restrictions being dictated by internal storage requirements. This then selects the classes of potential variables - material and construction modes, configurational variables and component sizes. In a sieve-search procedure, an attempt is made to consider all three classes.

In the particular case of SEV existing experience has indicated that a major portion of the structural design is governed by local hydrostatic and hydrodynamic pressure loadings. In addition, the requirement for internal cargo containers has a profound influence on the ranges of beam and bulkhead spacings which can be reasonably used in the ship design.

With these considerations, the design for minimum weight can be conducted on the basis of optimizing the structure for normal pressure loading and subsequently checking the resulting design for strength due to overall bending, shear and torsion loads. Plating (panel) thicknesses and beam cap areas are then increased to ensure the overall integrity of the structure. This approach led to two main procedural items - overall ship weight minimization and plating optimization. These led naturally to definition of the following variables:

a) Construction module, including both material and constructional characteristics. Figure 10 presents sixteen combinations of materials and constructions which were considered feasible for this type of system.

b) Configurational Variables (Figure 9)

   1) Longitudinal bulkhead spacing, l_{LB}

   2) Transverse bulkhead spacing, l_{TB}

c) Dimensional Variables

   1) Plating - Panel Skin Thickness and Stiffener Dimensions

A finite number of longitudinal and transverse bulkheads and transverse frame spacings are specified and these configurational variables are optimized for minimum weight. Optimization of the dimensional variables results in generation of the data banks which store pre-optimized dimensional variables of structural components. In the present application, panels of the type shown in Figure 11 were optimized for minimum weight on the basis of normal pressure. A penalty function formulation with a Rosenbrock[26] search procedure was used. Geometric programming

Figure 8. Prototype SEV

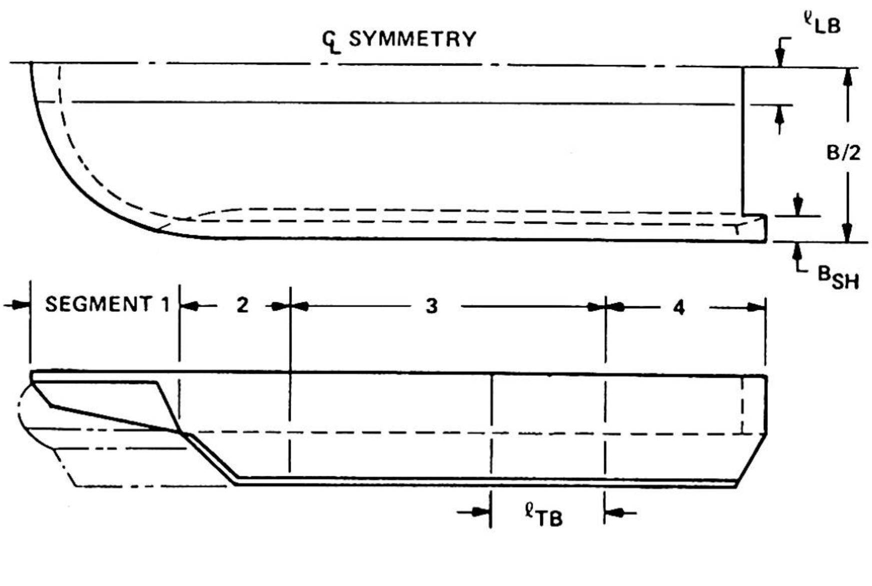


Figure 9. SEV Structural Envelope

methods for structural optimization[27] also appear extremely promising for use in component design where constraint and merit functions are expressible as nonlinear polynomials. The governing equations for the strength and stability of the panels under the action of uniaxial compression and in-plane shearing were also derived for use in the sieve-search method. For purposes of simplification, the panels were assumed infinitely wide and all critical conditions were expressed in terms of panel length, normal pressure, material characteristics and panel cross-sectional dimensions. The optimizations were then performed using the panel cross-sectional dimensions as variables. In addition to strength and stability constraints, consideration was also given to fabricational limitations for the various types of sections optimized.

Sixteen data banks consisting of eight basic geometric configurations with four materials namely, aluminum, steel, titanium, glass reinforced plastics were calculated and labeled construction modules. For these, all practical "failure" modes were derived analytically in five basic categories: material strength, overall buckling, local buckling, deformation limits and fabrication limits. Actually deformation and fabrication limits are not failure modes, but rather design specification modes which in many cases determined the optimum panel design.

When performing the optimization procedure, all of these critical conditions were expressed as inequality constraints. The fabricational constraints were based upon:

1) Considerations of practical sections, for example, no overlap of flanges, and

2) Data on the range of extruded sections which could be manufactured using existing dies and presses.

The deformation constraints were based upon the specific maximum allowable panel deflections.

The data banks are entered during the sieve-search process using the current spacing, L, and panel pressure, p, as shown by the dashed line on Figure 11. The resultant minimum weight, w, and cross-sectional geometry is stored for subsequent weight calculations.

A flow chart for the sieve-search program proper is shown in Figure 12. For application of the method, the vehicle was broken down into the four segments shown in Figure 9. These segments were defined in the present case by variations in the pressure loadings acting on the hull. Other forms of segmentation could have been selected to suit any arbitrary conditions. Within each segment certain configurational parameters were kept constant, although varying from segment to segment. The location of the longitudinal bulkheads was common to all segments. Each segment was further broken down into smaller zones such as deck, sidehull, etc. Each zone is then designed separately and combined to form the design of a segment.

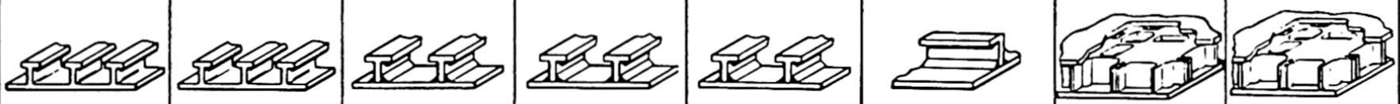
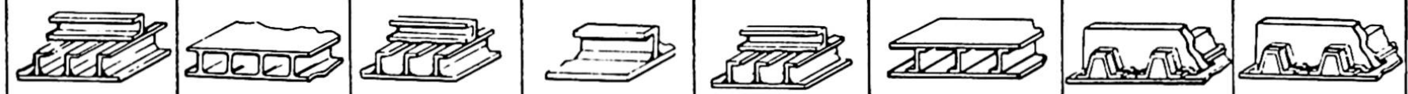
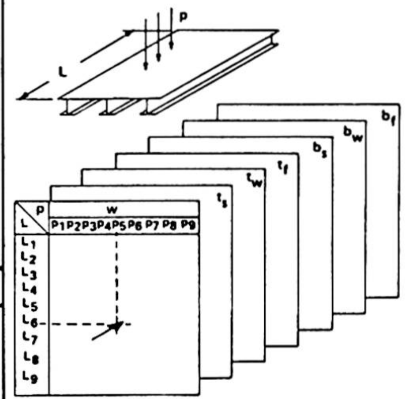
	Aluminum		Steel		Titanium		GRP	
Alloys	5086	6061	HY-130	ALMAR-362	6Al-2Cb-1Ta-1Mo	6Al-4V	Polyester, Fiberglass Reinforced	Epoxy, Fiberglass Reinforced
Temper or Condition	H-111 Extrusion H-34 Sheet H-117 Plate	T6 { Sheet Plate Extrusion	Q & T } Plate Extrusion	Q & T } Plate Extrusion	Annealed { Sheet Plate Rolled - Forms Extrusion	Annealed { Sheet Plate Rolled - Forms Extrusion	Mechanical - Hand Lay-up	Mechanical - Hand Lay-up
Type of Construction	 Stringers and Plate Integral Extrusion Welded Assy.	Stringers and Plate Integral Extrusion Riveted Assy.	Plate and Extrusion Welded Welded Assy.	Plate and Extrusion Welded Welded Assy.	Plate and Extrusion Welded Welded Assy.	Multiple Frames Plate and Rolled Forms Welded Assy.	Box Core Bonded Assy.	Box Core Bonded Assy.
Module	1(a)	1(b)	4(a)	4(b)	4(c)	5(b)	8(a)	8(b)
Type of Construction	 Formed Stringers and Frame - Sheet Welded Assy.	Extruded Web-Core Riveted Assy.	Formed Stringers and Frame - Sheet Welded Assy.	Multiple Frames Plate and Rolled Forms Welded Assy.	Formed Stringers and Frame - Sheet Welded Assy.	Sheet and Rolled Forms Resistance Welded Assy.	Stringers - Frames and Skin Bonded Assy.	Stringers - Frames and Skin Bonded Assy.
Module	2(a)	3	2(b)	5(a)	2(c)	6	7(a)	7(b)

Figure 10.Construction Modules



OPTIMIZATION PARAMETERS: p, L
OPTIMIZATION VARIABLES: w, t_s, t_w, t_f, b_s, b_w, b_f
OPTIMIZATION CONSTRAINTS: STRESSES DEFLECTION AND FABRICATIONAL

Figure 11.SEV Data Bank

The cyclic nature of the design process is apparent from the flow chart (Figure 12). It can be seen to be essentially a series of iterative looping operations which indeed permit the sequential consideration of all feasible possibilities.

The design process begins by selection of the appropriate SEV gross weight and construction module. The next choice is of the longitudinal bulkhead spacing from a list of allowable spacings. In SEV's cargo container size provides a lower bound on bulkhead spacing. For the specific longitudinal bulkhead spacing, allowable ranges of transverse bulkhead spacings are defined for each segment. In each segment the geometry is fully defined. Using the known local pressure loadings, the data banks are accessed for appropriate loads and geometry for each zone. The weight of a segment is computed and compared with that obtained for other transverse bulkhead spacings. This is repeated for each segment yielding the minimum weight design for the specified longitudinal bulkhead spacing. The entire looping is then carried out again for the next longitudinal bulkhead spacing and repeated to obtain the minimum weight craft.

Final checks on strength are performed using engineering analyses and where necessary incremental material is provided. For the ship system costing data is also computed.

The program then automatically cycles to the next construction module and SEV configuration, and repeats the entire process.

The above program was used extensively in the design of a range of SEV's varying from 500 to 10,000 tons gross weight.

Out of a total possible number of 232 ship designs, 173 were obtained. Designs for the remaining 59 configurations were not obtained due to the non-existence of minimum weight data for certain pressure/length combinations in the data banks. The availability of such data is directly dependent on the constraints placed on panel deflection, stress, and geometry in the process of generating the data banks. The constraints will yield, at times, nonfeasible panel designs and these appear as blanks in the data banks. If some of the constraints used in the component design are considered to be artificially severe, they may be modified. Using these less stringent criteria, additional ships designs would have been obtained.

Computational time was as low as 20 cpu seconds per ship design on an IBM 360/65 computer. The resulting output gave a very full description of the proposed structure including all scantlings, frame spacings and cost data.

As a second example of the use of the sieve-search procedure in a structural design process, its potential application to a bridge design problem is briefly considered.

For the purposes of a design study, a complete bridge structure may be broken down into the three major subdivisions,

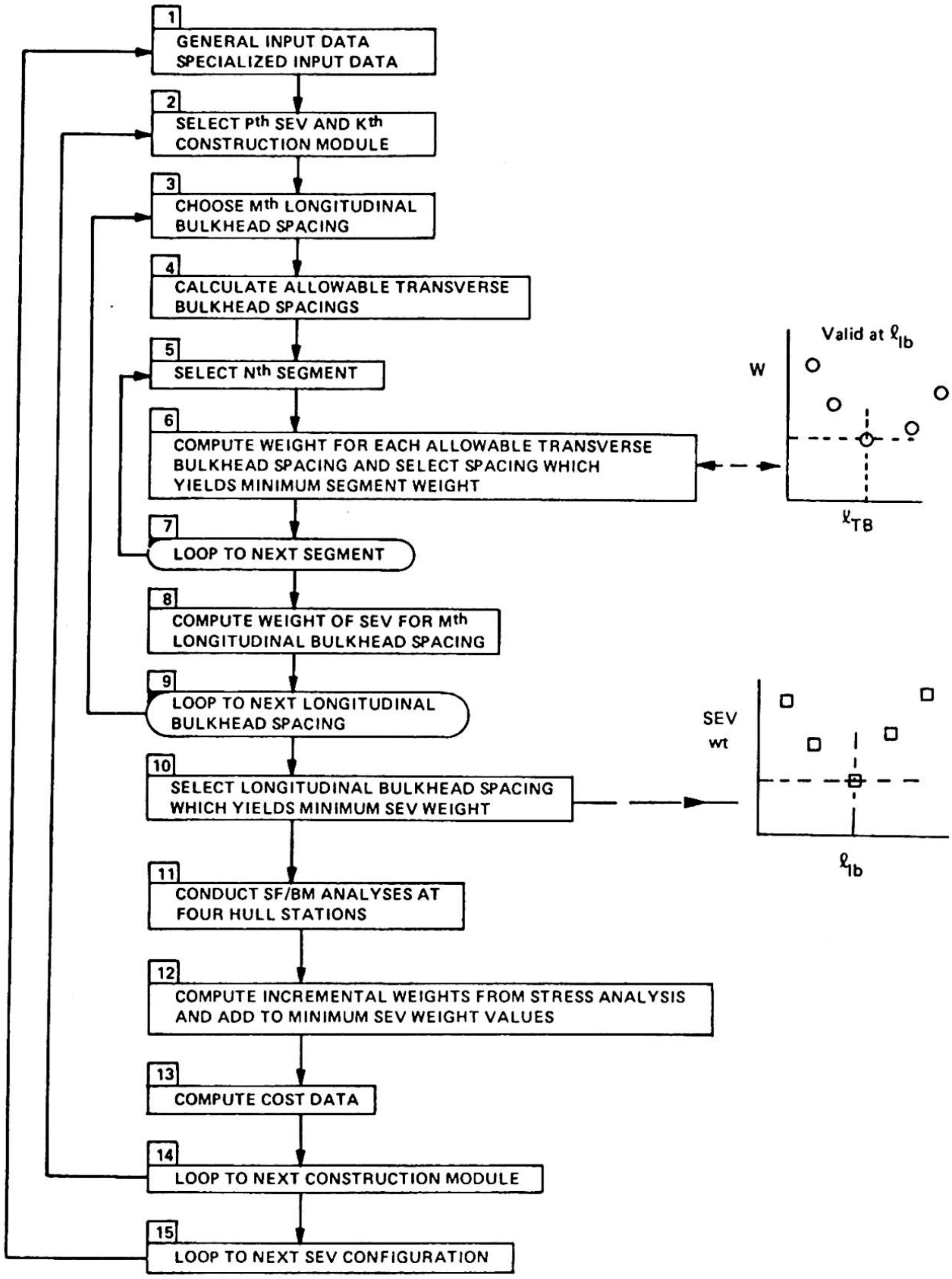


FIGURE 12. Sieve-Search for Minimum SEV Weight

(i) Deck

(ii) Primary structure spanning between piers and supporting the deck

(iii) Substructure

The use of a sieve-search procedure for the optimal design of the deck and primary structure is outlined in a flow chart (Figure 13).

The essential characteristics of multiple levels of iterations with detailed design performed through the use of data banks is retained from the previous example, although the actual operations performed at each iterative stage may be totally different. For the bridge example the use of multiple data banks is deemed necessary.

The data banks for a bridge structure may contain a variety of different construction modules such as deck panels, plate or tubular girders, precast concrete beams, steel wide-flange beams with cover plates, cable arrangements or steel towers and concrete columns. All such potential bridge structural components may be pre-optimized on any suitable merit basis for suitable ranges of critical loadings and span lengths. The optimized data is then stored in banks readily accessible at the appropriate stage of the sieve-search program.

In selecting the bridge configuration a number of choices may exist and each may be programmed according to its intrinsic shape. Table I from Reference 28 indicates that for various spans alternate configurations may be possible, but engineering judgment and/or environmental conditions as well as other factors may narrow the choice of feasible designs.

For the deck construction, the most commonly used constructions are in-situ concrete, precast concrete and steel. Also experiencing growing popularity is the so-called orthotropic steel deck consisting of deck plate stiffened by parallel stringers. Some typical cross-sections may be found in Reference 28. In order to choose an appropriate deck, the following prime factors must be considered,

1. Strength, longitudinal and transverse
2. Dead weight
3. Cost

An efficient design includes the deck as part of the primary structure for load transferal and the true economic evaluation of the above three items may be successfully achieved when and only when the total bridge design is considered. For example, an orthotropic steel deck if viewed only as a slab will not compete in cost with reinforced concrete but the steel deck may be competitive if its axial force capacity and reduced dead load effects are considered through the complete superstructure and substructure designs.

The comparisons of all typical deck sections in context with the complete bridge structure are ideally suited for an automatic

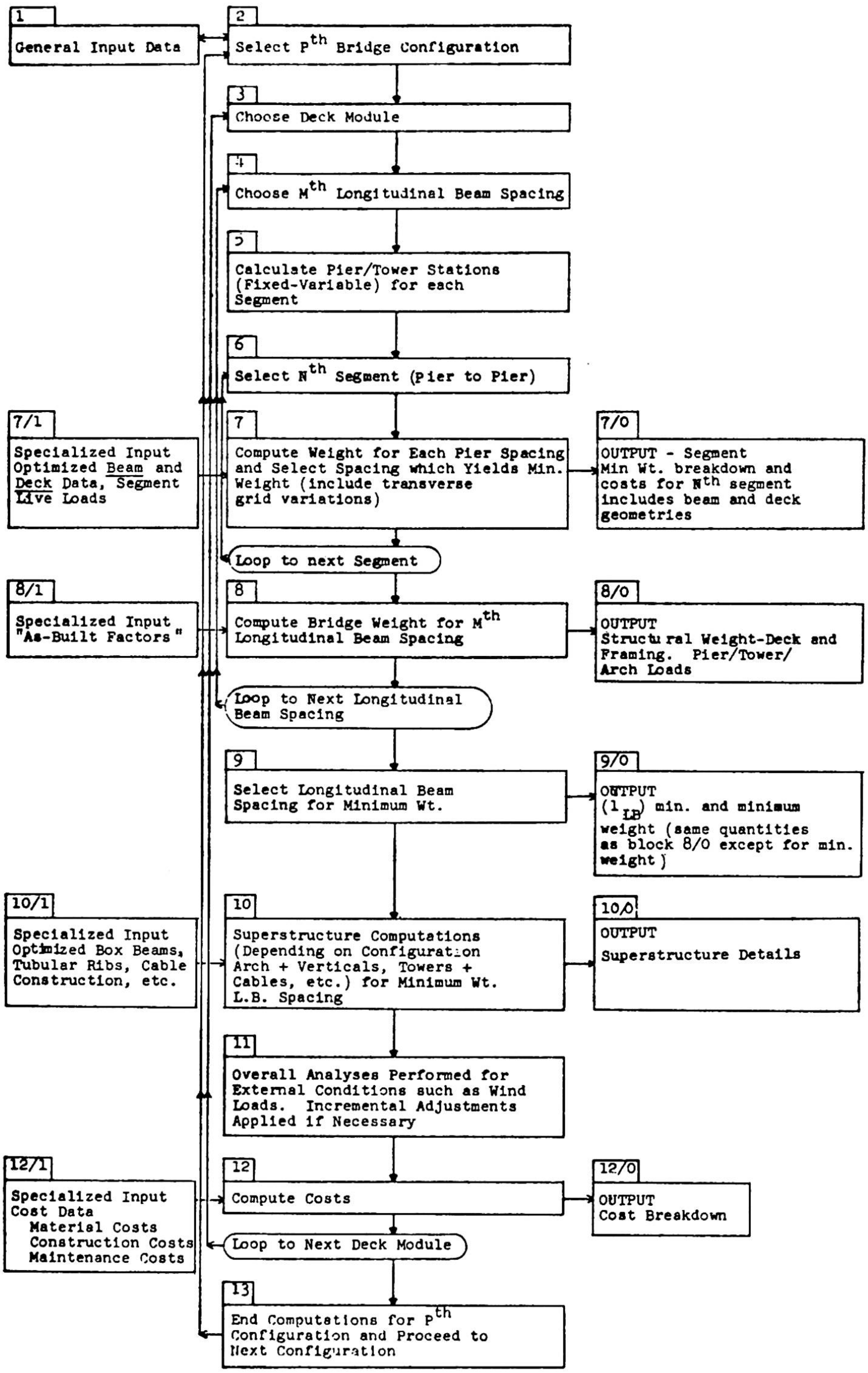


Figure 13. SIEVE-SEARCH FLOW CHART FOR BRIDGE DESIGN

sieve-search procedure. Data banks for each of the candidate deck cross-sections may be established. These files can be as sophisticated as desired wherein a range of span lengths together with a range of critical loads may be applied to each typical section. Figure 11 illustrates this and associated with each minimum weight (live + deal load), W_i, is a unit cost factor and optimal cross section geometries. The minimum weight and/or cost is evaluated under such constraints as deflection, strength, buckling, torsion, web crippling, etc. The definition here may be either working-load or ultimate. Fabricational limitations, code specifications and cost penalty factors may be included as well.

In the sieve-search, a predetermined table of acceptable longitudinal beam spacings may be specified, along with appropriate transverse spacings. The program will cycle through all the defined grids in its search for the optimum design. The configuration is also controlled by combination of fixed and variable lengths between abutments and piers. Each of the variable spans would be designated as a semi-independent segment for which a detailed design would be performed. For each segment, deck module and beam arrangement, the appropriate specialized data banks would be accessed to generate the local design which would then be compared with the previously stored optimal design. All segmental data is then assembled for the evaluation of the total design for a given longitudinal beam spacing. Specialized input, labeled "as-built" factors are provided to account for nonstructural items such as expansion joints, catwalks, railings, wearing surfaces, protective coatings, etc. After all potential longitudinal beam spacings have been considered, an interim optimal design is obtained. For this configuration, the superstructure is designed, again using appropriate specialized data banks. At this point a complete deck and superstructure have been designed and final check analyses should be performed. Some incremental adjustments on component sizing may be necessary. Consideration may even be given to the use of some suitable form of optimality-criteria optimization to refine a design, if this is felt to be appropriate.

Finally, the program would generate complete cost data for the selected design, including maintenance. The program is then repeated for other deck modules and configurations until the final design is rendered.

The preceding discussion has not been based upon an existing program but has been intended to indicate the possible extension of the sieve-search procedure to a civil engineering structure.

IV. CONCLUDING REMARKS

Two distinctly different approaches to the optimal design of structures have been presented. In both cases, the greatest possible emphasis has been placed on the practical aspects of the design problem in an attempt to produce a workable tool for the designer.

The optimality criteria approach is gaining acceptance by designers because of its fortuitous combination of simplicity and effectiveness. Computer programs based thereon are being used simply because no other method exists at this time that can cope with the very large number of variables encountered in finite element representations of real structures.

The use of the sieve-search procedure is a direct contrast in approach. The results obtained from the SEV design studies for an extremely modest expenditure of computer time, have indicated that this method is also an efficient cost-effective approach to automated optimal design. The ideal solution would possibly appear to be a combination of the two approaches, whereby the sieve-search defines configuration and noncontinuous variables and the optimality criteria method is used for refinement of the design. The extension of the procedures to other classes of design offers a considerable potential for overall system optimization.

V. REFERENCES

1. Gellatly, R. A. and Gallagher, R. H., "Development of Advanced Structural Optimization Programs and Their Application to Large Order Systems", Conf. on Matrix Methods in Structural Mechanics, Dayton, AFFDL-TR-66-80, 1966.

2. AGARD, "Symposium on Structural Optimization" Conference Proceedings, AGARD CP-36, Istanbul, Turkey, October 1969.

3. AGARD, "Second Symposium on Structural Optimization", Conference Proceedings AGARD CP-123, Milan, Italy, April 1973.

4. Pope, G. G. and Schmit, L. A., "Structural Design Applications of Mathematical Programming Techniques", AGARDograph No. 149, February 1971.

5. Gellatly, R. A. and Berke, L., "Optimal Structural Design", AFFDL-TR-70-165.

6. Gellatly, R. A., Dupree, D. M., Berke, L., "OPTIM II: A MAGIC Compatible Large Scale Automated Minimum Weight Design Program", AFFDL-74-97, Vol 1, July 1974.

7. AGARD, "Lecture Series on Structural Optimization", AGARD LS-70, October 1974.

8. Schmit, L. A., Farshi, B., "Some Approximation Concepts for Structural Synthesis," AIAA/ASME/SAE 14th Structures. Structural Dynamics and Materials Conference, Williamsburg, March 1973.

9. Batt, J. R., Eckman, E. and Dupree, D. M., "Joint Surface Effect Ship Program, Structural Design Studies," Phase I & II, Bell Aerospace Company Report No. 7363-950001,2, December 1970.

10. Gallagher, R. H., 'Fully-Stressed Design', Chapter in "Optimum Structural Design, Theory and Applications," edited by R. H. Gallagher and O. C. Zienkiewicz, John Wiley and Sons, 1973.

11. Razani, R., "The Behavior of the Fully-Stressed Design of Structures and its Relationship to Minimum Weight Design," AIAA Journal Vol. 3, No. 12, Dec. 1965, pp. 2262-2268.

12. Kicher, T. P., "Optimum Design-Minimum Weight Versus Fully-Stressed," Journal of Structural Division, ASCE, Vol. 92 No. ST6, Proc. Paper 5014, Dec. 1966, pp. 265-279.

13. Klein, B. "Direct Use of External Principles in Solving Certain Optimizing Problems Involving Inequalities," Journal, Operations Research Soc. of America, Vol. 3, No. 2, May 1955, pp. 168-175 and 548.

14. Schmit, L. A., "Structural Design by Systematic Synthesis," Proceedings of the Second Conference on Electronic Computation, ASCE, September 1960.

15. Venkayya, V. B., "Design of Optimum Structures," Computers and Structures, Vol. 1, 1971, pp. 265-309.

16. Nagtegaal, J. C., Div. of Eng., Brown Univ., "A New Approach to Optimal Design of Elastic Structures."

17. Venkayya, V. B., Khot, N. S., and Reddy, V. S., "Energy Distribution in an Optimum Structural Design," AFFDL-TR-68-156.

18. Dwyer, W., Emerton, R., and Ojalvo, I., "An Automated Procedure for the Optimization of Practical Aerospace Structures," AFFDL-TR-70-118.

19. Sheu, C. Y., Prager, W., "Recent Developments in Optimal Structural Design," Applied Mechanics Reviews, Vol. 21, No. 10, Oct. 1968.

20. Barnett, R. L. and Hermann, P. C., "High Performance Structures," NASA-CR-1038, 1968.

21. Berke, L., "An Efficient Approach to the Minimum Weight Design of Deflection Limited Structures," AFFDL-TM-70-4-FDTR, July, 1970.

22. Berke, L. and Khot, N. S., "Use of Optimality Methods for Large Scale Systems," AGARD LS-70, 1974.

23. Khot, N. S., Venkayya, V. B. and Berke, L., "Optimization of Structures for Strength and Stability Requirements," AFFDL-TR-73-98.

24. Venkayya, V. B., Khot, N. S., Tischler, V. A. and Taylor, R. F., "Design of Optimum Structures for Dynamics Loads", 3rd Conference on Matrix Methods in Structural Mechanics, Dayton, AFFDL-TR-71-160, 1973.

25. Taylor, J. E., and Lie, C. Y., "Optimal Design of Columns," AIAA J., Vol. 6, No. 8, pp. 1497-1502, August 1968.

26. Rosenbrock, H. H., "An Automatic Method for Finding the Greatest or Least Value of a Function," Computer Journal, Vol. 3, 1960.

27. Templeman, A. B., "The Use of Geometric Programming Methods for Structural Optimization," AGARD LS-70, 1974.

28. O'Connor, Colin, "Design of Bridge Superstructures," Wiley-Interscience, New York, 1971.

SUMMARY

Examples are presented of two approaches to the optimal design of complex structural systems. The first approach, based upon the use of optimality criteria is capable of optimizing finite element representations of large scale, complex structures with prescribed geometry. Both strength and stiffness constraints are considered. The second procedure is labeled sieve-search and is used for the overall optimization of structures. The method permits the full variation of construction method, materials and configuration as well as component sizing.

RESUME

Des exemples de calcul d'optimisation pour des systèmes de structures complexes sont présentées selon deux approches. La première, basée sur le critère d'optimisation, permet de résoudre des ensembles de grande dimension d'éléments finis, ou des structures complexes à géométrie donnée. Les contraintes de résistance et de raideur sont prises en considération. La seconde méthode, dite "sieve-search" (tamiser-chercher), sert à l'optimisation globale des structures. La méthode permet une complète variation de la méthode de construction, des matériaux, de la forme et des dimensions.

ZUSAMMENFASSUNG

Beispiele des Computer-Einsatzes bei der Optimierung von komplizierten Tragwerken sind nach zwei Methoden aufgeteilt. Die erste Methode wird das Optimierungskriterium benützen, und erlaubt die Optimierung von komplexen Tragwerken mit einer bestimmten Geometrie, durch mächtigen Darstellungen finiten Elementen. Die zweite Methode, die "sieve-search" (sieben-suchen) heisst, wird für die globale Optimierung von Tragwerken benützt. Sie erlaubt eine totale Bearbeitung der Baumethode, der Materialien, der Form und der Abmessungen.

Leere Seite
Blank page
Page vide