MINIMAX MULTIOBJECTIVE OPTIMIZATION IN STRUCTURAL DESIGN

C. H. TSENG AND T. W. LU

Department of Mechanical Engineering, National Chiao Tung University, Hsinchu, Taiwan, 30050 R.O.C.

SUMMARY

The use of multiobjective optimization techniques, which may be regarded as a systematic sensitivity analysis, for the selection and modification of system parameters is presented. A minimax multiobjective optimization model for structural optimization is proposed. Three typical multiobjective optimization techniques—goal programming, compromise programming and the surrogate worth trade-off method—are used to solve such a problem. The application of multiobjective optimization techniques to the selection of system parameters and large scale structural design optimization problems is the main purpose of this paper.

INTRODUCTION

In most structural optimization problems, it is desirable that the structure should be of minimum weight and simultaneously meet some requirements such as strength, deflection and frequency. Usually, these requirements are imposed as inequality constraints, where the allowable strength, deflection and frequency are selected as fixed parameters in the problem formulation.[1] Parameter sensitivity analysis is thus needed if the requirements are modified after the optimization is complete. Interest toward this topic can be seen in the literature.[2,3] It is noted that parameter analysis provides only a measure of changes of the optimum design within its neighbourhood. Another approach to treat the selection and modification of problem parameters is the use of multiobjective optimization techniques which may be regarded as a systematic sensitivity analysis.[4]

A general multiobjective optimization problem is to find the vector of design variables $\mathbf{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ which minimizes a vector objective function $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x}))^{\mathrm{T}}$ over the feasible design space $\mathbf{X}$. It is the determination of a set of non-dominated solutions (or Pareto optimum solutions, non-inferior solutions or efficient solutions) that achieves a compromise among several different, usually conflicting, objective functions. The set of non-dominated solutions is a set of solutions in which no decrease can be obtained in any of the objectives without causing a simultaneous increase in at least one of the other objectives. The associated set of non-dominated objectives generally represents a collection of incomparable solutions for the objectives which are usually non-commensurable to begin with. Thus, here arises the decision making problem from which a partial or complete ordering of the set of non-dominated objectives is accomplished by considering the preferences of the decision maker (DM). Most of the multiobjective optimization techniques are based on how to elicit the preferences and determine the best-compromise solution.

0029–5981/90/141213–16$08.00

Received 8 August 1989
Revised 22 March 1990

In this paper, a general multiobjective optimization model for structural optimization is presented. The traditional minimum weight problem can be reformulated as a optimization problem with multiple minimax objectives to consider the significant design requirements simultaneously and be treated as the parameter selection problem. Multiobjective optimization methods available in the literature are mostly linear in nature.[5] Only a few non-linear methods are proposed with simple testing problems. The application of multiobjective optimization techniques to large scale problems is still rare. Three typical multiobjective optimization techniques, goal-programming, compromise programming and the surrogate worth trade-off method, are used to solve large scale structural systems with multiple minimax objectives. The determination of system parameters by means of multiobjective optimization process is also discussed here.

PROBLEM FORMULATION

A common minimum weight structural optimization problem may be stated as the following.

Find the design variables $\mathbf{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ to minimize the weight of the structure $f(\mathbf{x})$ and satisfy stress constraints

$$|\sigma_i| \leqslant \sigma_{ia} \qquad i = 1, 2, \ldots, n \tag{1}$$

displacement constraints

$$|z_j| \leqslant z_{ja} \qquad j = 1, 2, \ldots, r \tag{2}$$

buckling constraints

$$\sigma_i \leqslant \sigma_i^{\mathrm{b}} \qquad i = 1, 2, \ldots, n \tag{3}$$

fundamental frequency constraints

$$\xi_{\mathrm{a}} \leqslant \xi \tag{4}$$

and design variable bounds

$$x_{il} \leqslant x_i \leqslant x_{iu} \tag{5}$$

where $\mathbf{x}$ can be the cross sectional area, length, width, ..., etc, of each member; σ_{ia} and z_{ja} are the allowable stress for the ith member and allowable displacement for the jth displacement component, respectively; σ_i is the stress in the ith member; z_j is the jth displacement component. The allowable buckling stress σ_i^{b} is obtained from the equation of the Euler critical load. ξ_{a} is the allowable eigenvalue calculated from the specified lower limit on the fundamental frequency. The fundamental frequency ω can be calculated by solving the following eigenvalue problem:

$$\mathbf{K}(\mathbf{x})\mathbf{y} = \xi\mathbf{M}(\mathbf{x})\mathbf{y} \tag{6}$$

where $\mathbf{M}(\mathbf{x})$ is the $r \times r$ mass matrix, $\mathbf{y}$ is an eigenvector and ξ is an eigenvalue which equals ω^2. For detailed information of the derivation and calculation, references can be consulted in References 1 and 6.

Once the problem is formulated, a non-linear programming algorithm is exploited to perform the design optimization. But it should be noted that the specification of allowable parameters could be hard to select for the engineer. In fact, one may originally hope to minimize the weight and nodal displacements as well as member stresses and maximize the fundamental natural frequency simultaneously, which lead to a multiobjective optimization problem. All these objectives are non-commensurable and conflict in nature. From this point of view, it is desirable to relax the original single objective optimization problem to the multiobjective one.

Engineers and the decision makers will communicate to make a best-compromise design during the multiobjective optimization solution process. This would be beneficial to the balance among the specified objectives. As a result, we redefine the structural optimization problem as the following.

Find the design variable **x** which minimizes the weight of the structure, the maximum member stress and the maximum nodal displacement while the fundamental natural frequency is maximized. Mathematically, it can be stated as

$$\min f_1(\mathbf{x}) \tag{7}$$

$$\min f_2(\mathbf{x}) = \max \sigma_i, \quad i = 1, 2, \ldots, n \tag{8}$$

$$\min f_3(\mathbf{x}) = \max z_j, \quad j = 1, 2, \ldots, r \tag{9}$$

$$\min f_4(\mathbf{x}) = -\omega \tag{10}$$

subject to

$$\sigma_i \leqslant \sigma_i^{\mathrm{b}} \qquad i = 1, 2, \ldots, n \tag{11}$$

$$x_{il} \leqslant x_i \leqslant x_{iu} \tag{12}$$

Since the buckling strength of each member is σ_i^{b}, determined once the geometric property is given, the buckling constraint will not be treated as an objective here. With these objectives, one may be able to choose the system parameters by the solution of the multiobjective optimization problem, even for large scale ones. For example, a material with a strength greater than or equal to the maximum stress would be an appropriate selection for the structure. On the other hand, the maximum displacement should be reduced to fit the requirement of the performance of the structure. A higher fundamental frequency is preferred for the sake of vibration control. All the information obtained during the multiobjective optimization solution process is useful for the determination of allowable parameters of structural systems.

In the above general formulation, the selection of objectives depends on the DM and the engineer. Only those the DM considers worthwhile or whose performance is not certain to the engineer will appear in the set of objectives. Since most of the structural optimization problems in the real world are large scale ones, we have to deal with large scale structural optimization problems with multiple minimax objectives. Through the design of three structures, we attempt to show how multiobjective optimization techniques perform in the selection of system parameters as a systematic parameter sensitivity analysis and how they are adopted to solve large scale design problems.

SOLUTION PROCEDURES

A set of multiobjective optimization techniques which have been or may be applied to structural design problems was presented in References 4, 7 and 8. Generally, multiobjective optimization techniques can be classified into three categories according to the articulation of the decision maker's preference structure over the multiple objectives prior to, during or after the optimization. Based on this classification, an overall review of multiobjective optimization techniques and their applications is available in Reference 9. Three typical multiobjective optimization techniques—goal programming, compromise programming and the surrogate worth trade-off method—are thus chosen in this study, mainly because of their representative and thier promising features for future development in engineering design optimization.

Goal programming

Goal programming, first presented in References 5 and 7, requires prior articulation of preferences by the DM. It allows the decision maker to specify a target for each objective function. A preferred solution is then defined as the one that minimizes the sum of the absolute values of deviations from the prescribed set of goal values.

A general formulation of goal programming for non-linear programming problems can be stated as follows:[10]

$$\min f(\mathbf{x}) = \left\{ \sum_{i=1}^{k} (d_i^+ + d_i^-)^p \right\}^{1/p}; \quad p \geqslant 1 \tag{13}$$

subject to

$$\begin{aligned}
&g_i(\mathbf{x}) \leqslant 0, && i = 1, 2, \ldots, m\\
&h_i(\mathbf{x}) = 0, && i = 1, 2, \ldots, p\\
&f_i(\mathbf{x}) - d_i^+ + d_i^- = T_i, && i = 1, 2, \ldots, k\\
&d_i^+ \geqslant 0, && i = 1, 2, \ldots, k\\
&d_i^- \geqslant 0, && i = 1, 2, \ldots, k
\end{aligned}$$

where T_i represents the target or goal set by the DM for the ith objective function $f_i(\mathbf{x})$ and d_i^+ and d_i^- are, respectively, the under-achievement and over-achievement of the ith objective and defined as

$$d_i^+ = 0{\cdot}5\{|f_i(\mathbf{x}) - T_i| + (f_i(\mathbf{x}) - T_i)\} \tag{14}$$

$$d_i^- = 0{\cdot}5\{|f_i(\mathbf{x}) - T_i| - (f_i(\mathbf{x}) - T_i)\} \tag{15}$$

Note that the value of p is based on the utility function chosen by the DM and it is possible to obtain non-dominated solutions by varying the values of p and T_i.

Compromise programming

Compromise programming[11] is based on the concept of distance from an ideal solution, and the solutions which are closest to the ideal point are called compromise solutions and form the compromise set. When each objective function f_i, $i = 1, \ldots, k$, is characterized by 'more is better', one can set $\mathbf{f}^* = (f_1^*, \ldots, f_k^*)$, where $f_i^* = \sup\{f_i(\mathbf{x})|\mathbf{x} \in \mathbf{X}\}$. In this case, $\mathbf{f}^*$ is called an ideal point because usually $\mathbf{f}^*$ is not attainable. Compromise was once remarked as the art of cutting the cake so that everyone thinks he or she got the largest piece. Here, compromise is considered as an effort to approach the ideal solution as closely as possible. The ideal solution, usually infeasible, is the situation where everybody does get the largest possible piece, though in compromise it is assumed that the cake is always smaller than the sum of claims on it.

The generalized distance measure used in compromise programming to evaluate how close the set of non-dominated points come to the ideal point is the family of L_s matrices defined as

$$L_s = \left\{ \sum_{i=1}^{k} \alpha_i^s \left| \frac{f_i(\mathbf{x}) - f_i^*}{f_{i\,\max} - f_i^*} \right|^s \right\}^{1/s} \tag{16}$$

where $1 \leqslant s \leqslant \infty$; α_i are weights, f_i^* and $f_{i\,\max}$ are the minimum value and the worst value for the ith objective respectively; $f_i(\mathbf{x})$ is the value of implementing the design variable $\mathbf{x}$ with respect to the ith objective. Therefore, the compromise programming model is formulated as follows:

$$\begin{aligned} &\min L_s(\mathbf{x}) = L_s(\mathbf{x}_s^*) \\ &\text{subject to } \mathbf{x} \in \mathbf{X} \end{aligned} \tag{17}$$

for a given set of weights α_i and for all $1 \leqslant s \leqslant \infty$, where $\mathbf{X}$ denotes the feasible region.

To find a compromise solution, one needs to know the ideal point $\mathbf{f}^* = (f_1^*, \ldots, f_k^*)$ and the parameter s for the $L_s(\mathbf{x})$ matrices. For $s = 1$, all deviations from f_i^* are taken into consideration in proportion to their magnitude. For $2 \leqslant s < \infty$, the larger the deviation, the larger the weight in L_s. As s become larger and larger, and finally $s = \infty$, the largest deviation is the only one taken into account.

The surrogate worth trade-off method

The surrogate worth trade-off (SWT) method[5, 12, 13] is an interactive method, which can be applied when design variables are continuous and objective functions and constraints are twice differentiable. It assumes that the preference of the decision maker is monotonic and can be exhibited as an implicit multiattribute utility function as the DM systematically compares two objectives at a time. The SWT method consists of three main steps. First of all, generate a representative subset of non-dominated solutions called 'promising solutions' and obtain the relevant trade-off information for each generated solution. Secondly, interact with the DM to elicit the preference (worth) structure. Finally, select the best-compromise solution from the information obtained.

Consider the ε-constraint problem $P_i(\varepsilon)$

$$\begin{aligned} &\min && f_i(\mathbf{x}) \\ &\text{subject to} && f_j(\mathbf{x}) \leqslant \varepsilon_j, \quad j \neq i, \quad j = 1, 2, \ldots, k \\ &&& \mathbf{x} \in \mathbf{X} \end{aligned} \tag{18}$$

where $\mathbf{X}$ is the original design space of the multiobjective optimization problem and ε_j are known as the maximum tolerable levels. It is used in the method to yield the promising solutions and associated trade-off information. By use of the concept of duality theory and Lagrange multipliers as well as the Kuhn–Tucker conditions, the trade-off function T_{ij} can be defined as

$$T_{ij} = \frac{\partial f_i(\mathbf{x})}{\partial f_j(\mathbf{x})} \tag{19}$$

or

$$T_{ij} = -\lambda_{ij} \tag{20}$$

λ_{ij} is the generalized Lagrange multiplier associated with the ith ε-constraint. More details of the derivation of the trade-off function can be seen in the Appendix and References 12 and 13. The trade-off λ_{ij} represents the marginal rate of substitution between the ith and jth objectives. Note that only positive trade-offs are of interest here. Once the trade-offs have been identified, the decision maker is supplied with the trade-off information and then asked to express his ordinal preference. The DM's preference is embodied in the surrogate worth function w_{ij}, which is ordinal and ranges from -10 to $+10$. It is an estimation of the desirability of the trade-off λ_{ij}. Thus, $+10$ would indicate that λ_{ij} marginal units of the ith objective are very much more worthwhile than one marginal unit of the jth objective, while -10 would indicate the opposite and 0 would signify an even trade. The best-compromise solution is determined when all trade-off ratios are selected to make the surrogate worth functions simultaneously equal to 0.

Numerical implementation

Since the techniques for the single (scalar) objective optimization problem are well established, most multiobjective optimization techniques are based on transforming multiobjective problems into single objective ones in order to obtain the best-compromise solution by applying single objective optimization techniques directly. From the comparative studies of various single optimization algorithms available in the recent literature,[14-18] the mathematical programming method known as sequential quadratic programming (SQP) is selected in this paper for its accuracy, reliability and efficiency. The auxiliary single objective optimization problems needed during the solution process are solved by using the SQP algorithm in the optimization solver IDESIGN3.51. The IDESIGN3.51 program with a subroutine format is modified from IDESIGN3.5.[19] It has been robustly implemented and extensively tested in a wide range of highly non-linear small and large scale problems for the SQP algorithm.[20]

One typical approach to a single minimax objective is the solution of the minimization of an introduced artificial variable which is the upper bound of the maximum value of the objective. For minimax multiobjective optimization problems, they can be treated along the way of the solution of a single objective problem to have more than one artificial variable. In the SWT method, for the trade-off interpretation in the minimax multiobjective problem one may refer to Chankong and Haimes.[13]

The ideal point is selected as the goal in goal programming. Since the ideal/worst point is the point at which each individual objective is minimized/maximized without regard to other objectives, the over-achievement constraint in equation (13) will be eliminated. But to find an ideal/worst point is not always attainable from the numerical point, which can be provided by the DM during the solution process. All the information of ideal and worst point for both goal and compromise programming may be calculated in the implemented program or given by the DM from input data.

For the analysis of the structures, the finite element method is used. And the necessary design sensitivity analysis is performed by the hybrid method,[21] which combines the advantage of the direct differential method (DDM) and the adjoint variable method (AVM). The implementation of the three techniques is used for the solution of three large scale numerical design examples.

DESIGN EXAMPLES

Three structural optimization problems frequently appearing in the literature are reformulated as multiobjective optimization problems, as discussed previously. They are, as shown in Figures 1–3, a ten-member cantilever truss, a twenty-five-member transmission tower and a two-hundred-member plane truss.[1] Three design problems are imposed, all constrained. The ten-member cantilever truss is solved with loading case II. The twenty-five-member transmission tower design problem is considered under two loading conditions and the 200-bar truss is treated with three loading conditions. All the numerical data are the same as those in Reference 1, while the allowable parameters are abandoned and will be determined during the multiobjective optimization solution (decision making) process.

Tables I–V shows the best-compromise solutions and trade-off information for the three example problems. Note that the ideal value and worst value of maximum stress, displacement and minus frequency are provided by the DM or the engineer. The range between the ideal and worst values may be associated with the properties of available materials or requirements of the design. Trade-offs between the objectives prior to, during or after the solution process are

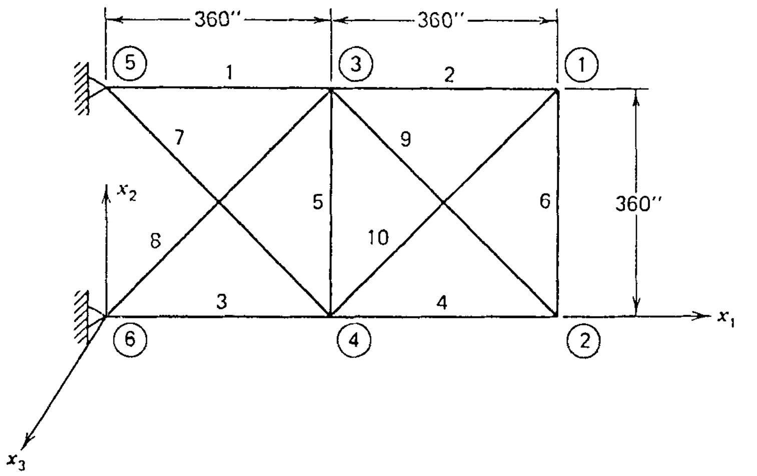


Figure 1. Ten-member cantilever truss

Table I. Results for the ten-member cantilever truss by (i) goal programming; (ii) compromise programming

	Goal programming		Compromise programming	
	$p = 1$	$p = 2$	$s = 1$	$s = 2$
Design $\mathbf{x}^*$, in^2				
x_1	2·514E1	2·509E1	6·879E1	6·522E1
x_2	1·071E0	9·984E − 1	1·094E0	1·085E0
x_3	2·615E1	2·651E1	5·632E1	5·380E1
x_4	1·299E1	1·294E1	3·038E1	2·888E1
x_5	1·000E − 1	1·000E − 1	1·000E − 1	1·000E − 1
x_6	1·972E0	2·111E0	2·454E0	2·455E0
x_7	1·328E1	1·397E1	2·378E1	2·365E1
x_8	1·566E1	1·560E1	6·228E1	5·939E1
x_9	1·728E1	1·714E1	3·871E1	3·702E1
x_{10}	1·000E − 1	1·000E − 1	1.000E − 1	1.000E − 1
Objective $\mathbf{f}^*$				
Weight, lb	4·785E3	4·822E3	1·209E4	1·157E4
Max. stress, psi	2·491E1	2·332E1	2·000E1	2·000E1
Max. displ., in	2·000E0	2·000E0	7·890E − 1	8·275E − 1
Frequency, Hz	− 2.200E1	− 2·200E1	− 2·500E1	− 2·496E1
Under-achievements				
d_1^+	2·000	39·000	—	—
d_2^+	4·910	3.320	—	—
d_3^+	1·500	1.500	—	—
d_4^+	3·000	3.000	—	—
CPU time (sec)	18·0	14·0	39·0	30·0

Weight: min = 4·783E3, max = 2·941E4
Max. stress: min = 20·0, max = 25·0
Max. displacement: min = 0·5, max = 2·0
Frequency: min = − 25·0, max = − 22·0

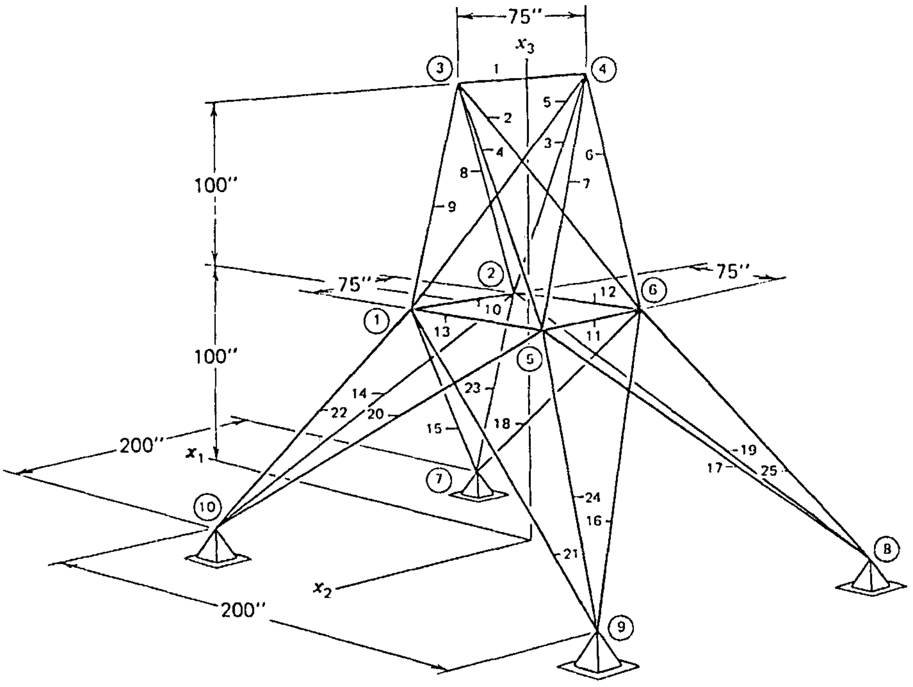


Figure 2. Twenty-five-member transmission tower

Table II. Trade-off information and the best-compromise solution for the ten-member cantilever truss design problem by the SWT method assuming the existence of a decision maker

f_1	f_2	f_3	f_4	λ_{12}	λ_{13}	λ_{14}	w_{12}	w_{13}	w_{14}
Ideal value:									
4·783E3	20·0	0·5	− 25·0						
Worst value:									
2·941E4	25·0	2·0	− 22·0						
Referenced objective:			Weight (f_1)						
Maximum tolerable levels:									
	22·5	1·2	− 23·0						
Trade-off									
1 9·98E3	2·15E1	9·20E − 1	− 2·38E1	1·17E2	9·66E3	4·89E3	0	+ 3	− 2
2 9·34E3	2·15E1	9·90E − 1	− 2·40E1	1·33E2	8·99E3	5·09E3	0	0	+ 2
3 9·32E3	2·15E1	9·90E − 1	− 2·39E1	1·28E2	8·99E3	4·87E3	0	0	0

	Best-compromise solution			
Objectives	f_1	f_2	f_3	f_4
	9·3192E3	2·1500E1	9·9000E − 1	− 2·3900E1

Designs									
1	2	3	4	5	6	7	8	9	10
55·775	1·147	4·964	2·420	0·1	2·279	18·447	38·470	31·887	0·1

CPU time (sec)	55·0

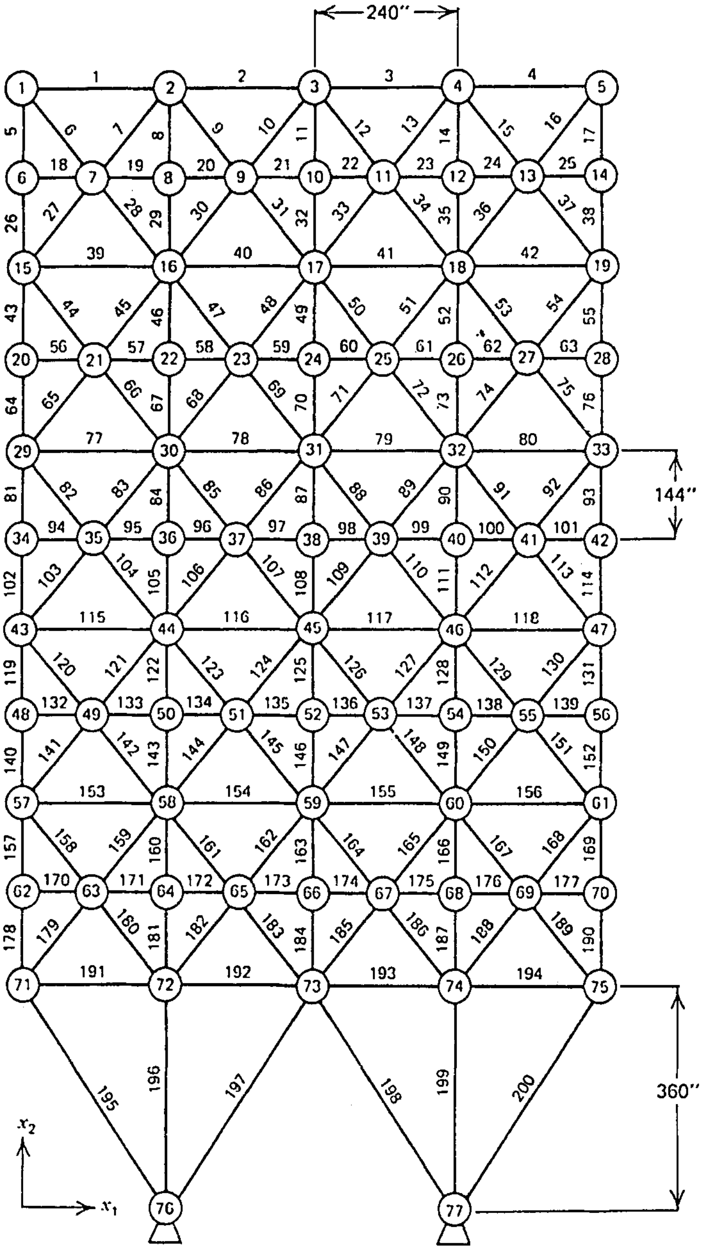


Figure 3. Two-hundred-member plane truss

accomplished by communication of the DM or the engineer to the program. Resources and constraints affect the solution process largely. For example, materials available for the design should be taken into consideration when the trade-off between maximum stress and other objectives proceeds. Maximum displacement directly affects the performance of the structure. And weight represents the cost, while fundamental frequency is important for excitation vibration. All these objectives are non-commensurable and may conflict with each other. After the

Table III. Results for the twenty-five-member transmission tower design problem by (i) goal programming; (ii) compromise programming

	Goal programming		Compromise programming	
	$p = 1$	$p = 2$	$s = 1$	$s = 2$
Member no.				
1	1·000E − 1	1·000E − 1	1·000E − 1	1·577E0
2, 3, 4, 5	2·175E0	2·971E0	1·464E1	1·436E1
6, 7, 8, 9	2·407E0	3·209E0	1·714E1	1·481E1
10, 11, 12, 13	1·000E − 1	3·132E − 1	9·685E − 1	1·000E − 1
14, 15, 16, 17	1·109E0	1·091E0	7·462E0	8·368E0
18, 19, 20, 21	2·035E0	2·764E0	1·503E1	1·678E1
22, 23, 24, 25	2·963E0	4·268E0	2·331E1	2·143E1
Objective **f***				
Weight, lb	6·062E2	8·094E2	4·401E3	4·363E3
Max. stress, psi	3·500E1	3·500E1	3·500E1	3·500E1
Max. displ., in	3·425E − 1	2·538E − 1	4·862E − 2	5·045E − 2
Frequency, Hz	− 7·750E1	− 7·750E1	− 7·992E1	− 7·992E1
Under-achievements				
d_1^+	12·10	215·3	—	—
d_2^+	0·000	0·000	—	—
d_3^+	0·333	0·244	—	—
d_4^+	2·500	2·500	—	—
CPU time (sec)	54·0	54·0	67·0	44·0

Weight: min = 5·941E2, max = 3·131E4
Max. stress: min = 35·0, max = 40·0
Max. displacement: min = 0·01, max = 0·35
Frequency: min = − 80·0, max = − 77·5

best-compromise solution is reached, system parameters can be determined according to the solution. It is noted that design alternatives are compared systematically during the solution process while traditional parameter sensitivity analysis yields only local information after the optimization.

Goal programming gives maximum values for maximum displacement and frequency objectives with $p = 1$ and $p = 2$ in 10-bar truss design problem, as shown in Table I. Frequency objective achieves the worst point for all the cases except $p = 1$ in 200-bar case. In the 25- and 200-bar truss design problems, the maximum stress equals the ideal value. All this information is clear with the investigation of under-achievements. It should be noted that a solution is a dominated solution if the under-achievements are all zero.

The last row of the tables shows the CPU time needed on Apollo DN 570. Evidently the 200-bar truss design case requires much CPU time. Generally, compromise programming requires more CPU time than goal programming does. It produces higher weight and lower maximum displacement for all the cases. This is the scaling and weighting effect, both of weight and maximum displacement.

For goal and compromise programming, the ideal point is selected as the goal. However, the DM may choose his favourite point as his goal and the balance between the objectives may be in terms of the value of p and s or the weightings α. So the best-compromise solution would be different for other DMs. This can be seen in the solution process of the SWT method.

Table IV. Trade-off information and the best-compromise solution for the twenty-five-member transmission tower design problem by the SWT method assuming the existence of a decision maker

	f_1	f_2	f_3	f_4	$\nabla f_2 \frac{\partial \mathbf{x}(\varepsilon)}{\partial \varepsilon_l}$	λ_{13}	λ_{14}	w_{13}	w_{14}
Ideal value:									
	5·941E2	35·0	0·01	− 80·0					
Worst value:									
	3·131E4	40·0	0·35	− 77·5					
Referenced objective:			Weight (f_1)						
Maximum tolerable levels:									
		38·0	0·2	− 78·5					
Trade-off									
1	1·68E3	3·70E1	1·24E − 1	− 7·92E1	0·0	1·68E3	—	+ 6	—
2	1·46E3	3·70E1	1·43E − 1	− 7·92E1	0·0	—	2·04E3	—	+ 2
3	1·46E3	3·70E1	1·43E − 1	− 7·90E1	0·0	1·46E3	1·76E3	+ 5	0
4	1·04E3	3·70E1	2·00E − 1	− 7·90E1	0·0	1·04E3	1·26E3	0	0

Best-compromise solution

Objectives	f_1	f_2	f_3	f_4
	1·042E3	3·700E1	2·000E − 1	− 7·895E1

Designs (Member group)						
1	2	3	4	5	6	7
0·1	3·796	4·017	0·590	1·184	3·550	5·828
CPU time (sec)	3187·0					

Table V. Results for the two-hundred-member plane truss design problem by (i) goal programming; (ii) compromise programming

	Goal programming		Compromise programming	
	$p = 1$	$p = 2$	$s = 1$	$s = 2$
Objective **f***				
Weight, lb	2·941E4	3·580E4	1·518E5	1·582E5
Max. stress, psi	2·500E1	2·500E1	2·500E1	2·500E1
Max. displ., in	5·000E − 1	4·267E − 1	1·415E − 1	1·393E − 1
Frequency, Hz	− 5·000E0	− 6·444E0	− 1·000E1	− 9·745E0
Under-achievements				
d_1^+	0·000	0·639E4	—	—
d_2^+	0·000	0·000	—	—
d_3^+	0·456	0·382	—	—
d_4^+	5·000	3·556	—	—
CPU time (sec)	20 825·0	17 259·0	37 342·0	20 361·0

Weight: min = 2·941E4, max = 6·578E5
Max. stress: min = 25·0, max = 30·0
Max. displacement: min = 0·0445, max = 0·5
Frequency: min = − 10·0, max = − 5·0

Recall that we are dealing with positive Lagrange multipliers as the trade-offs. It is necessary to treat zero trade-offs λ_{ij} because of the improper selection of ε_j or the non-conflicting nature between the jth objective and the referenced objective. The situation may occur if there are several objectives. In the solution of 25- and 200-bar design problems, we have encountered such a case. It is noted that the given ε_j for the ε-constraint of the stress objective may not produce a positive trade-off rate because of the limitation of buckling constraints and the interaction with other objectives. To handle such zero λ_{ij}'s, we have to ask the DM the following question: 'How would you like to decrease f_k by λ^i_{kl} units and change f_j by $\nabla f_j(\mathbf{x}^i)\cdot\partial\mathbf{x}(\varepsilon^i)/\partial\varepsilon_l$ units, while increasing f_l by one unit? Given that $f_j = f_j(x^i)$ for all $j = 1, \ldots, n$.'

Note that determining $\partial x(\varepsilon^i)/\partial\varepsilon_l$ is not easy for large scaie problems, especially for structural optimization problems. This may be the main difficulty we have to overcome. Here in this section we use forward difference to compute $\partial\mathbf{x}(\varepsilon^i)/\partial\varepsilon_l$ as

$$\frac{\partial\mathbf{x}(\varepsilon^i)}{\partial\varepsilon_l} \approx \frac{\mathbf{x}(\varepsilon^i_l) - \mathbf{x}_i}{\delta\varepsilon_l} \tag{21}$$

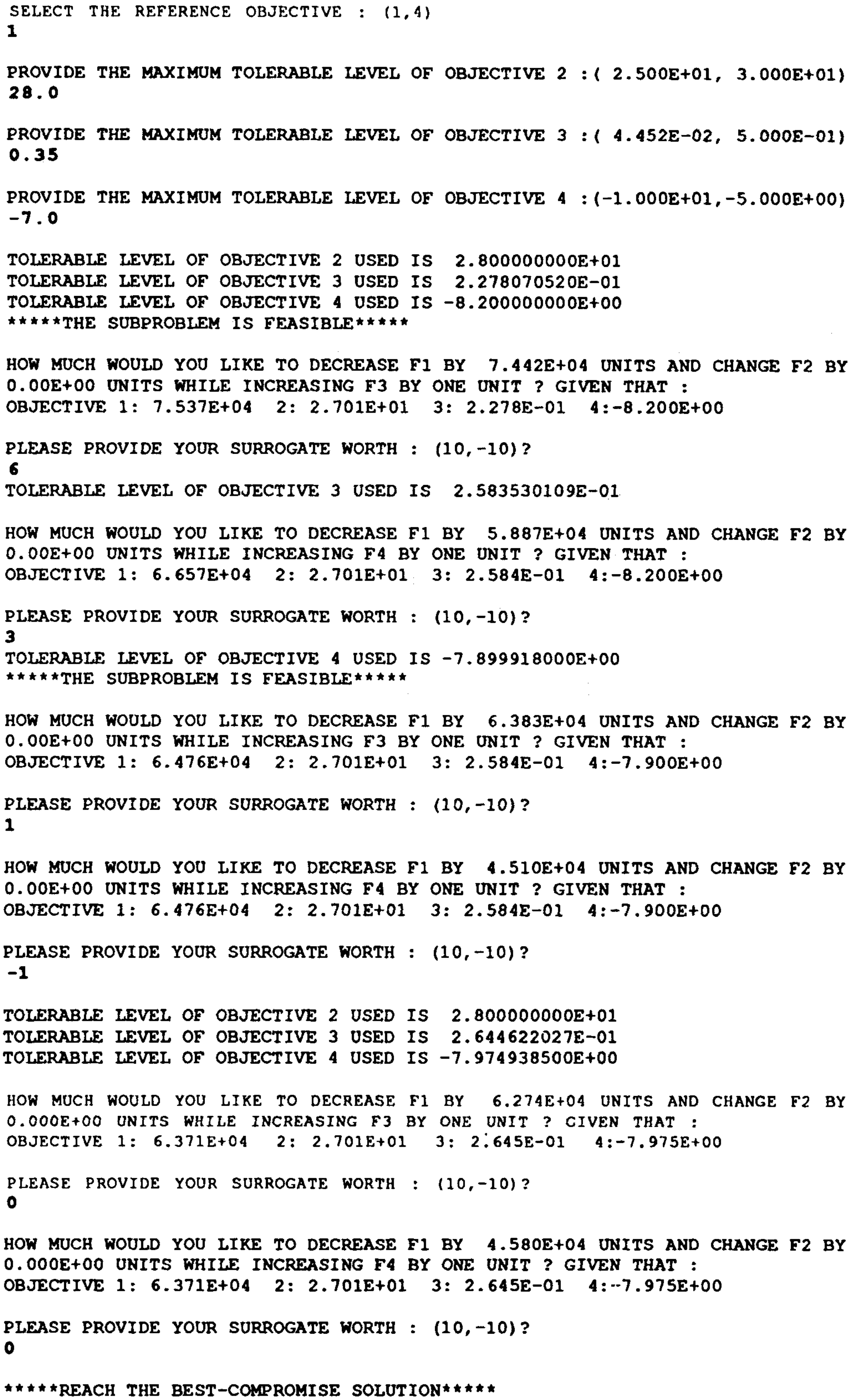
where $\varepsilon^i_l = (\varepsilon^i_1, \varepsilon^i_2, \ldots, \varepsilon^i_l + \delta\varepsilon_l, \ldots, \varepsilon^i_{k-1})$. As a result, we have to solve more auxiliary single objective optimization problems each time before the above question is asked, depending on the number of positive λ_{ij}'s. Moreover, the trade-off rate between minimax objectives should be noted.

Tables II, IV and VI display the summary of the trade-off information and the best-compromise solution obtained by the SWT method. From these results, we may realize that trade-offs of weight to maximum displacement and frequency are much more significant than to the maximum

Table VI. Trade-off information and the best-compromise solution for the two-hundred-member plane truss design problem by the SWT method assuming the existence of a decision maker

f_1	f_2	f_3	f_4	$\nabla f_2 \frac{\partial\mathbf{x}(\varepsilon)}{\partial\varepsilon_l}$	λ_{13}	λ_{14}	w_{13}	w_{14}
Ideal value:								
2·941E4	25·0	4·452E − 2	− 10·0					
Worst value:								
6·578E5	30·0	0·5	− 5·0					
Referenced objective:			Weight (f_1)					
Maximum tolerable levels:								
	28·0	0·35	− 7·0					
Trade-off								
1 7·54E4	2·70E1	2·28E − 1	− 8·20E0	0·0	7·44E4	—	+ 6	—
2 6·66E4	2·70E1	2·58E − 1	− 8·20E1	0·0	—	5·89E4	—	+ 3
3 6·48E4	2·70E1	2·58E − 1	− 7·90E1	0·0	6·38E4	4·51E4	+ 1	− 1
4 6·37E4	2·70E1	2·65E − 1	− 7·98E1	0·0	6·27E4	4·58E4	0	0

	Best-compromise solution			
Objectives	f_1	f_2	f_3	f_4
	6·371E4	2·701E1	2·645E − 1	− 7·975E1
CPU time (sec)	192 566·0			



```
SELECT THE REFERENCE OBJECTIVE : (1,4)
1

PROVIDE THE MAXIMUM TOLERABLE LEVEL OF OBJECTIVE 2 :( 2.500E+01, 3.000E+01)
28.0

PROVIDE THE MAXIMUM TOLERABLE LEVEL OF OBJECTIVE 3 :( 4.452E-02, 5.000E-01)
0.35

PROVIDE THE MAXIMUM TOLERABLE LEVEL OF OBJECTIVE 4 :(-1.000E+01,-5.000E+00)
-7.0

TOLERABLE LEVEL OF OBJECTIVE 2 USED IS  2.800000000E+01
TOLERABLE LEVEL OF OBJECTIVE 3 USED IS  2.278070520E-01
TOLERABLE LEVEL OF OBJECTIVE 4 USED IS -8.200000000E+00
*****THE SUBPROBLEM IS FEASIBLE*****

HOW MUCH WOULD YOU LIKE TO DECREASE F1 BY  7.442E+04 UNITS AND CHANGE F2 BY
0.00E+00 UNITS WHILE INCREASING F3 BY ONE UNIT ? GIVEN THAT :
OBJECTIVE 1: 7.537E+04  2: 2.701E+01  3: 2.278E-01  4:-8.200E+00

PLEASE PROVIDE YOUR SURROGATE WORTH : (10,-10)?
6
TOLERABLE LEVEL OF OBJECTIVE 3 USED IS  2.583530109E-01

HOW MUCH WOULD YOU LIKE TO DECREASE F1 BY  5.887E+04 UNITS AND CHANGE F2 BY
0.00E+00 UNITS WHILE INCREASING F4 BY ONE UNIT ? GIVEN THAT :
OBJECTIVE 1: 6.657E+04  2: 2.701E+01  3: 2.584E-01  4:-8.200E+00

PLEASE PROVIDE YOUR SURROGATE WORTH : (10,-10)?
3
TOLERABLE LEVEL OF OBJECTIVE 4 USED IS -7.899918000E+00
*****THE SUBPROBLEM IS FEASIBLE*****

HOW MUCH WOULD YOU LIKE TO DECREASE F1 BY  6.383E+04 UNITS AND CHANGE F2 BY
0.00E+00 UNITS WHILE INCREASING F3 BY ONE UNIT ? GIVEN THAT :
OBJECTIVE 1: 6.476E+04  2: 2.701E+01  3: 2.584E-01  4:-7.900E+00

PLEASE PROVIDE YOUR SURROGATE WORTH : (10,-10)?
1

HOW MUCH WOULD YOU LIKE TO DECREASE F1 BY  4.510E+04 UNITS AND CHANGE F2 BY
0.00E+00 UNITS WHILE INCREASING F4 BY ONE UNIT ? GIVEN THAT :
OBJECTIVE 1: 6.476E+04  2: 2.701E+01  3: 2.584E-01  4:-7.900E+00

PLEASE PROVIDE YOUR SURROGATE WORTH : (10,-10)?
-1

TOLERABLE LEVEL OF OBJECTIVE 2 USED IS  2.800000000E+01
TOLERABLE LEVEL OF OBJECTIVE 3 USED IS  2.644622027E-01
TOLERABLE LEVEL OF OBJECTIVE 4 USED IS -7.974938500E+00

HOW MUCH WOULD YOU LIKE TO DECREASE F1 BY  6.274E+04 UNITS AND CHANGE F2 BY
0.000E+00 UNITS WHILE INCREASING F3 BY ONE UNIT ? GIVEN THAT :
OBJECTIVE 1: 6.371E+04  2: 2.701E+01  3: 2.645E-01  4:-7.975E+00

PLEASE PROVIDE YOUR SURROGATE WORTH : (10,-10)?
0

HOW MUCH WOULD YOU LIKE TO DECREASE F1 BY  4.580E+04 UNITS AND CHANGE F2 BY
0.000E+00 UNITS WHILE INCREASING F4 BY ONE UNIT ? GIVEN THAT :
OBJECTIVE 1: 6.371E+04  2: 2.701E+01  3: 2.645E-01  4:-7.975E+00

PLEASE PROVIDE YOUR SURROGATE WORTH : (10,-10)?
0

*****REACH THE BEST-COMPROMISE SOLUTION*****
```

Figure 4. An illustration of the interactive solution process for the optimization of structures with multiple minimax objectives by the SWT method assuming the existence of a decision maker (200-bar case)

stress. This agrees with those obtained by goal and compromise programming methods. Figure 4 is an illustration of the interactive solution process for the optimization of structures with multiple minimax objectives. During the interactive process, the DM or the engineer can systematically compare the objectives from the trade-off information provided by the program and reach the best-compromise solution.

CONCLUSION

The use of a minimax multiobjective optimization model to provide a systematic parameter sensitivity analysis is discussed. Three structural optimization problems are reformulated and solved by three multiobjective optimization techniques. Among these methods, goal and compromise programming need less computational effort but the trade-off between objectives is much difficult than for the SWT method. This is because the SWT method uses first order information of the objectives. If the DM or the engineer has already had a favourite goal in mind, goal programming may be used. On the other hand, compromise programming will be chosen if the computational efforts are emphasized and the specific goal is not provided in advance. The SWT method gives a very good procedure for the systematic parameter sensitivity analysis though it requires much more computational effort. Numerical experience of the SWT method shows that the optimum point for each auxiliary single objective optimization problem may be taken as the initial guess for the next one. Such a treatment may significantly save computational effort.

The selection of the system parameters before the optimization is difficult. It is even more difficult for large scale problems. This paper solved the problem and completed the application of multiobjective optimization techniques to large scale structural design optimization problems.

ACKNOWLEDGEMENT

The research reported in this paper is supported under a project sponsored by the National Science Council Grand, Taiwan, R.O.C., NSC78-0401-E009-12.

APPENDIX

Trade-off functions

For the multiobjective optimization problem formulated as

$$\min f_i(\mathbf{x}), \qquad i = 1, 2, \ldots, k \tag{A1}$$

subject to

$$g_i(\mathbf{x}) \leqslant 0, \quad i = 1, 2, \ldots, m \tag{A2}$$

the trade-off function, denoted $T_{ij}(\mathbf{x})$, between the ith and jth objectives, is defined as

$$T_{ij}(\mathbf{x}) = \frac{\mathrm{d}f_i(\mathbf{x})}{\mathrm{d}f_j(\mathbf{x})} \tag{A3}$$

where

$$\mathrm{d}f_i(\mathbf{x}) = \sum_{k=1}^{n} \frac{\partial f_i(\mathbf{x})}{\partial x_k} \mathrm{d}x_k$$

It is evident that the direct use of the definition is impractical and prohibitive. Thus, the concept of duality theory and Lagrange multipliers in non-linear programming as well as the ε-constraint method are used as the alternative approach to construct the trade-off matrix.

The ε-constraint problem $P_i(\varepsilon)$ can be considered as

$$\min f_i(\mathbf{x}) \tag{A4}$$

subject to

$$f_j(\mathbf{x}) \leqslant \varepsilon_j, \quad j \neq i, \quad j = 1, 2, \ldots, k \tag{A5}$$

$$g_l(\mathbf{x}) \leqslant 0, \quad l = 1, 2, \ldots, m \tag{A6}$$

where $\varepsilon_j = f_j^*(\mathbf{x}) + \varepsilon'_j, j \neq i, j = 1, 2, \ldots, k, \varepsilon'_j \geqslant 0, j \neq i, j = 1, 2, \ldots, k. f_i^*(\mathbf{x})$ are the minimum solution of the ith objective of problem (A1) and ε'_j will be varied parametrically in the process of constructing the trade-off functions. ε_j are known as the maximum tolerable levels. The generalized Lagrangian L of problem (A4) is

$$L = f_i(\mathbf{x}) + \sum_{j=1}^{m} \mu_j g_j(\mathbf{x}) + \sum_{\substack{j=1 \\ j \neq i}}^{k} \lambda_{ij} [f_j(\mathbf{x}) - \varepsilon_j] \tag{A7}$$

where μ_k and λ_{ij} are the generalized Lagrange multipliers. The behaviour of the minimum value of the objective function as ε_j varies is of special interest, so we have

$$\frac{\partial L}{\partial \varepsilon_j} = -\lambda_{ij} \tag{A8}$$

or

$$\lambda_{ij} = -\frac{\partial L}{\partial \varepsilon_j} \tag{A9}$$

Applying the Kuhn–Tucker conditions,

$$\lambda_{ij} \geqslant 0, \quad \lambda_{ij}(f_j(\mathbf{x}) - \varepsilon_j) = 0, \quad j \neq i, \quad i, j = 1, 2, \ldots, k \tag{A10}$$

$$\mu_l \geqslant 0, \quad \mu_l g_l = 0, \quad l = 1, 2, \ldots, m \tag{A11}$$

and $f_i(\mathbf{x}) = L$.

We find that equations (A10)–(A11) hold only if $\lambda_{ij} = 0$, or $f_j(\mathbf{x}) - \varepsilon_j = 0$, or both. Only non-zero Lagrange multipliers correspond to the non-dominated solutions, whereas the zero Lagrange multipliers correspond to the dominated solutions. It should be noted that only those $\lambda_{ij} > 0$, which implies $f_j(\mathbf{x}) = \varepsilon_j$, are important for our analysis. Therefore, equation (A9) becomes

$$\lambda_{ij} = -\frac{\partial f_i(\mathbf{x})}{\partial f_j(\mathbf{x})} \tag{A12}$$

or

$$T_{ij} = -\lambda_{ij} \tag{A13}$$

There exist the following relationships between λ_{ij}:

$$\lambda_{kj} = \lambda_{ki} \lambda_{ij} \quad \text{for all } k \neq j \tag{A14}$$

and

$$\lambda_{ki} = \frac{1}{\lambda_{ik}} \tag{A15}$$

The use of the trade-off function relationships can dramatically save computational effort.

REFERENCES

1. E. J. Haug and J. S. Arora, *Applied Optimal Design: Mechanical and Structural Systems*, Wiley, New York, 1979.
2. G. N. Vanderplaats and N. Yoshida, 'Efficient calculation of optimum design sensitivity', *AIAA J.*, **23,** 1798–1803 (1985).
3. T. J. Beltracchi and G. A. Gabriele, 'A RQP based method for estimating parameter sensitivity derivatives', *The 1988 ASME Design Technology Conferences—The Design Automation Conference, DE-VOL. 14,* ASME, New York, 1988, pp. 155–164.
4. J. Koski, 'Multicriterion optimization in structural design', in E. Atrek *et al.* (eds.), *New Directions in Structural Design*, Wiley, New York, 1984.
5. A. Goicoechea, D. R. Hansen and L. Duckstein, *Multiobjective Decision Analysis with Engineering and Business Applications*, Wiley, New York, 1982.
6. J. S. Arora, 'On improving efficiency of an algorithm for structural optimization and a user's manual for program trussopt3', *Technical Report No. 12,* Materials Engineering Division, The University of Iowa, Iowa City, IA 52242, 1976.
7. L. Duckstein, 'Multiobjective optimization in structural design: The model choice problem', in E. Atrek *et al.* (eds.), *New Directions in Optimum Structural Design*, Wiley, New York, 1984.
8. J. Koski and R. Silvennoinen, 'Pareto optima of isostatic trusses', *Comp. Methods Appl. Mech. Eng.*, **31,** 265–279 (1982).
9. C. H. Tseng and T. W. Lu, 'Multiobjective optimization in mechanical and structural design', *Math. Comp.*, submitted.
10. S. S. Rao, V. B. Venkayya and N. S. Khot, 'Optimization of actively controlled structures using goal programming techniques', *Int. j. numer. methods eng.*, **26,** 183–197 (1988).
11. M. Zeleny, *Multiple Criteria Decision Making*, McGraw-Hill, New York, 1982.
12. Y. Y. Haimes and W. A. Hall, 'Multiobjectives in water resource systems analysis: The surrogate worth trade off method', *Water Resources Res.*, **10,** 615–624 (1974).
13. V. Chankong and Y. Y. Haimes, *Multiobjective Decision Making: Theory and Methodology*, North-Holland, Amsterdam, 1983.
14. E. Sandgren and K. M. Ragsdell, 'The utility of nonlinear programming algorithms: A comparative study—Part I and II', *J. Mech. Des.*, **102,** 540–551 (1980).
15. W. Hock and K. Schittkowski, 'A comparative performance evaluation of 27 nonlinear programming codes', *Computing 30*, Springer-Verlag, Berlin, 1983, pp. 335–358.
16. A. D. Belegundu and J. S. Arora, 'A study of mathematical programming methods for structural optimization, Part I: Theory, Part II: Numerical aspects', *Int. j. numer. methods eng.*, **21,** 1583–1624 (1985).
17. P. B. Thanedar, J. S. Arora, C. H. Tseng, O. K. Lim and G. J. Park, 'Performance of some SQP algorithms on structural design problems', *Int. j. numer. methods eng.*, **23,** 2187–2203 (1986).
18. C. H. Tseng, J. J. Tsay, O. K. Lim, G. J. Park and J. S. Arora, 'Performance of several optimization algorithms on a set of nonlinear programming test problems', *Technical Report No. ODL-87.2,* Optimal Design Laboratory, College of Engineering, The University of Iowa, Iowa City, IA 52242, 1987.
19. C. H. Tseng and J. S. Arora, 'On implementation of computational algorithms for optimal, design 1: Preliminary investigation, design 2: Extensive numerical investigation', *Int. j. numer. methods eng.*, **26,** 1365–1402 (1988).
20. J. S. Arora and C. H. Tseng, 'IDESIGN user's manual version3.5', *Technical Report No. ODL-87.1,* Optimal Design Laboratory, College of Engineering, The University of Iowa, Iowa City, IA 52242, 1987.
21. C. H. Tseng and K. Y. Kao, 'Performance of a hybrid design sensitivity analysis on structural design problems', *Comp. Struct.*, **33,** 1125–1131 (1989).