Struct Multidisc Optim 23, 280–296
Digital Object Identifier (DOI) 10.1007/s00158-002-0186-2

Approximating the Pareto curve to help solve biobjective design problems

G. Fadel and Y. Li

Abstract When faced with multiple objectives, designers have to find ways to combine these objectives to find the solution that satisfies acceptable trade-off levels. In this paper, we present a methodology based on approximating the Pareto set of biobjective problems and presenting these trade-off graphs to the designer to facilitate decisions on trade-off. Once a solution is selected from the approximated set, the designer can select to either set a target on one or both objectives and use one of two methods to find the sought after solution. The paper details the methodology and applies it to three structural problems of increasing complexity, showing that the procedure provides also useful feedback even in the case of nonconvex Pareto sets.

Key words multi-objective optimization, Pareto-set, trade-off decisions, approximation, trusses

1 Introduction

The design process is in large part, a process of making decisions. The decisions are made based on the nature of the product, its potential market, its features, its cost, and other relevant issues. Furthermore, at different design stages, different decisions are called for. Researchers are currently investigating this critical aspect of the design process. They often use techniques developed in economics and mathematical sciences to learn how to deal with multi-objective or multi-criteria problems (in this paper both words are interchangeably used). Typically, decision making is the process of selecting a possible course from all the possible available alternatives. For most problems, the decision-maker (DM) wants to attain more than one objective or goal in selecting the results (or the course of action). When considering multiple objectives, the optimization process must include ways to combine those objectives, that is to find a certain super-objective function such that the results of this optimization problem with the super-objective function can satisfy the decision-maker's or the customer's preferences.

Traditionally, one approach for solving the multi-objective problem consists in using weights to combine the objective functions together to form the following problem:

$$\text{minimize} \sum_{i=1}^{k} w_i f_i(\underline{x})$$

subject to

$$g_{\,\mathrm{i}}(\underline{x}) \leq= 0; \quad i = 1, 2, \ldots, m\,,$$

$$\underline{x} = [x_1, x_2, \cdots, x_n] \;\in\; \mathrm{X} \subset R^n\,.$$

Usually, the weights are normalized so that their sum is equal to one. Initially and most commonly, a linear weighted method is used; the weights are assigned according to the relative importance of the objective functions. Once this weighting is imposed, the problem reverts to a single objective optimization problem that may be readily solved using traditional optimization techniques (Athan and Papalambros 1996).

Another commonly used method is to rank the relative importance of each objective function, optimize the most important objective function, put some predetermined acceptable values for all other objective functions, and append them to the constraint set of the optimization problem.

Both of the above formulations can be used to solve the multi-objective problem, many more have been suggested and used (Boychuk and Ovchinnikov 1973; Haimes and Chankong 1983; Keeney and Raiffa 1999; Das and

Received October 18, 2000

G. Fadel and Y. Li

Mechanical Engineering Department, Clemson University Clemson, SC 29634-0921, USA
e-mail: gfadel@ces.clemson.edu

Dennis 1997), and each is suitable for certain kinds of problems. The weighting method for instance is not indicated for nonconvex problems since in such a case, it can only find a subset of the Pareto points on the Pareto frontier (Koski 1985; Athan and Papalambros 1996; Messac *et al.* 2000). The ε-constraint method is similar to the second method described above, various constraints are set, and the Pareto points are generated. Goal programming attempts to reach targets for all the objectives simultaneously (Stadler 1988; Stadler and Dauer 1993; Li 1999).

In all these cases, the solution obtained may not be the best or the most satisfactory to the designer. Furthermore in many methods, the Decision-Maker (DM) needs to determine some weight for each objective function before the problem can be solved (a priori). In most cases there is not enough information to establish a set of weights which leads to "optimal" results. Some methods also put the burden on the designer to decide early on, on the "acceptable levels" of the various criteria. Other methods such as the physical programming method of Messac (1996) require the designer to establish ranges of desirability levels for the individual criteria.

Recent advances in design theory and decision based design resulted in a regain in popularity of multicriteria decision making in academic circles and in industry. Many researchers have recognized that decision based design involves multiple and often conflicting objectives (e.g. mass, stiffness, stress, deformation, stability in a structural problem) and should be treated in a multiobjective framework (Eschenauer 1992; Stadler and Dauer 1993). Others contend that a single objective should drive design, that of utility, and that all other "objectives" should be folded into that objective using a methodology such as the multi–attribute utility. In either case, knowledge about the trade-off issues will help formulate a preference and is thus a needed exercise.

The approach presented below applies to bicriteria optimization problems (BCOPs). As such it is restricted in application. However, the designer can use the method to approach a multi-criteria problem considering two objectives at a time (but may not necessarily reach the "best" solution). A generalization to more than two criteria is currently under investigation.

2 Methodology

The proposed methodology is based on approximating the biobjective Pareto set. For this approximation, the hyper-ellipse method described by Li and Fadel (1998) is used. Thus, the bicriteria or biobjective optimization programming problems can be solved as follows.

1. Formulate the problem as a BCOP problem

$$\min \begin{cases} f_1(\underline{x}) \\ f_2(\underline{x}) \end{cases}$$

$$\text{s.t.} \quad g_j(\underline{x}) \le 0\,, \quad j = 1, 2, \ldots, m\,,$$

$$\underline{x} = [x_1, x_2, \ldots, x_n] \quad \in \mathrm{X} \subset R^n\,. \tag{1}$$

2. Solve each objective independently, obtaining the ideal solution of each problem.
   Let x_1^{**} and x_2^{**} be the solutions of the above two single objective optimization problems and let $f_1^{\min} = f_1(x_1^{**})$, $f_1^{\max} = f_1(x_2^{**})$, $f_2^{\min} = f_2(x_2^{**})$, and $f_2^{\max} = f_2(x_1^{**})$. These four values form two points in the objective space, $(f_1^{\min}; f_2^{\max})$ and $(f_1^{\max}; f_2^{\min})$,

$$\min \ f_i(\underline{x}) \quad i = 1, 2\,,$$

$$\text{s.t.} \quad g_j(\underline{x}) \le 0\,, \quad j = 1, 2, \ldots, m\,,$$

$$\underline{x} = [x_1, x_2, \ldots, x_n] \ \in \mathrm{X} \subset R^n\,. \tag{2}$$

3. Using engineering judgement, select a suitable weight w for one of the objectives. Solve the following problem for the third point needed to determine the hyper-ellipse:

$$\min \quad w f_1(\underline{x}) + (1-w) f_2(\underline{x})\,,$$

$$\text{s.t.} \quad g_j(\underline{x}) \le 0\,, \quad j = 1, 2, \ldots, m\,,$$

$$\underline{x} = [x_1, x_2, \cdots, x_n] \ \in \mathrm{X} \subset R^n\,. \tag{3}$$

   Let the solution be x_3^{**}. The third point in the objective space is $[f_1(x_3^{**}); f_2(x_3^{**})]$ or f_1, f_2.
4. Using the three points obtained above, construct the hyper-ellipse

$$\left(\frac{f_1 - f_1^{\min}}{f_1^{\max} - f_1^{\min}}\right)^{\nu} + \left(\frac{f_2 - f_2^{\min}}{f_2^{\max} - f_2^{\min}}\right)^{\nu} = 1\,,$$

or

$$\left(\frac{f_1 - f_1^{\max}}{f_1^{\min} - f_1^{\max}}\right)^{\nu} + \left(\frac{f_2 - f_2^{\max}}{f_2^{\min} - f_2^{\max}}\right)^{\nu} = 1\,. \tag{4}$$

   The exponent ν needs to be solved iteratively.
5. Present the approximated Pareto set (the hyper-ellipse) to the decision-maker. Obtain the DM's preference in terms of the objective function value(s). Usually the results from optimization are normalized during the optimization process. The decision-maker might encounter some difficulty to understand the meaning of the normalized objective function values. So the approximation curve presented to the DM should be in its original scale. After the preference information (in the original scale) is obtained, it must be normalized before the procedure continues.
6. According to the DM's preference information, the following procedure can be used to obtain the values of the design variables.

   (a) If *one* of the objective function values is given, (f_1^* for instance), the ε-constrained method (Lieberman 1991; Li 1999), can be used to solve for the other objective function value and the values of the design variables. The given function value is used as the constraint for the corresponding objective function, $\varepsilon_1 = f_1^*$

$$\min\ f_2(\underline{x})\,,$$

$$\text{s.t.}\quad g_j(\underline{x}) \le 0\,,\quad j = 1, 2, \ldots, m\,,$$

$$f_1(\underline{x}) \le \varepsilon_1\,,$$

$$\underline{x} = [x_1, x_2, \ldots, x_n] \in \mathrm{X} \subset R^n\,. \tag{5}$$

(b) If *both* of the objective function values are given, f_1^* and f_2^*, goal programming (Charnes and Cooper 1977) can be used to solve for values of the deviation variables and the values of the design variables. The given function values are served as goals in the goal programming procedure.
Let $b_1 = f_1^*$ and $b_2 = f_2^*$, the following equations will yield the desired results:

$$\min \left|\sum_{j=1}^{k} \left(d_j^- + d_j^+\right)^p\right|^{1/p}\,,\quad p \ge 1\,,$$

$$\text{s.t.}\quad g_i(\underline{x}) \le 0\,,\quad i = 1, 2, \ldots m\,,$$

$$f_j(\underline{x}) + d_j^- - d_j^+ = b_j\,,\quad j = 1, 2\,,$$

$$d_j^-\ d_j^+ \ge 0\,,\quad \forall j\,,$$

$$d_j^- \cdot d_j^+ = 0\,,\quad \forall j\,. \tag{6}$$

The parameter p can be set according to the nature of the problem. Usually, $p = 1$ or 2. If unsure, let $p = 1$.

7. If the DM is not satisfied with the solution from step 6, repeat step 5 with other limits. Otherwise, the decision-making procedure ends.

Note that in the decision-making procedure, the approximated Pareto solution is provided to the decision-maker to make trade-off decisions. The decision-maker can obtain the value of the other objective when the first one is fixed. The approximation error is eliminated after the final optimization process.

The hyper-ellipse approximation of the Pareto set provides the decision-maker with a graphical illustration of the trade-off decisions and thus facilitates this process. This approximation of the Pareto curve also gives the sensitivity information of the objectives with respect to each other. This sensitivity information further helps the DM to make trade-off decisions. For example, if the value of the sensitivity derivative $|\mathrm{d}f_2/\mathrm{d}f_1|$ is very small, this means that at this point, any change of f_1 causes little change of f_2. Point A in Fig. 1 is one such point. One the other hand, point A is a good choice when the minimum of f_2 is desired. This point gives a relatively small value of f_2 while keeping the other objective function f_1 relatively small.

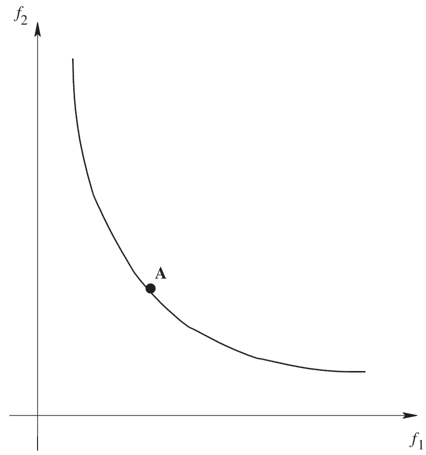

Fig. 1 Trade-off decision based on the Pareto curve

3 Applications

The degree of complexity of engineering applications varies greatly depending on the nature and the discipline of the application. Some applications are unsolvable without certain degrees of simplification. The truss problems are typical structural optimization problems. The degree of complexity of the truss problems depends on the number of bars and on the topology of the structure. By selecting the number of bars and designing the topology of the structure, the degree of complexity can be easily controlled. These problems are commonly used as examples in the optimization literature to verify the effectiveness and usability of new algorithms and methods without losing any generality. In the following section, three truss problems are discussed. The decision-making procedure presented earlier is applied to those applications. The results are analyzed in detail.

3.1 Two-bar truss problem

Consider the two-bar problem described by Chen (1995) and shown in Fig. 2. This structure consists of two bars. The truss members are pipes with average diameter d and thickness T in order to reduce the total weight. The commonly encountered formulation of this problem is to minimize the total weight (volume) of the structure subject to some stress constraint. This problem is statically determinate. All the internal forces and displacements can

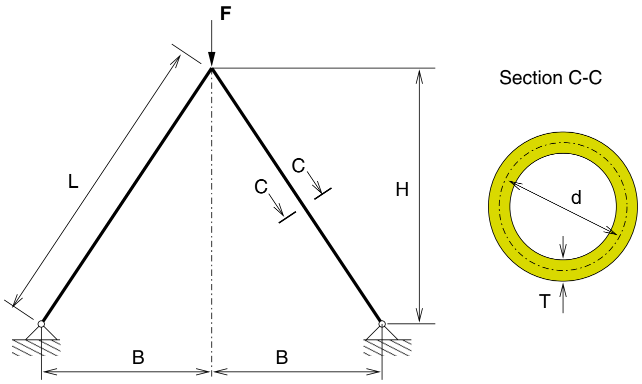

Fig. 2 The two-bar truss problem

be easily solved analytically. If there is a desire that both the volume and the normal stress be minimized, the problem has two criteria. To maintain integrity, the problem is restated below.

Let the design variable $x_1 = d$ and $x_2 = H$, d is the average diameter of the truss member and H is the height of the structure, the volume can be obtained using the following equation:

$$V = \Pi\left(R^2 - r^2\right)\sqrt{B^2 + H^2} \approx$$

$$2\Pi dT\sqrt{B^2 + H^2} = 2\Pi dx_1\sqrt{B^2 + x_2^2}\,. \tag{7}$$

The two-bar truss problem the normal stress can be obtained as

$$\sigma = \frac{S}{A} = \frac{F}{2\Pi dTH}\sqrt{B^2 + H^2} = \frac{F}{2\Pi dx_1 x_2}\sqrt{B^2 + x_2^2}\,. \tag{8}$$

Here, S is the internal force of the corresponding truss member. The constraints are applied to the maximum normal stress constraint

$$\sigma \leq \sigma_{\text{allowable}}\,, \quad \sigma \leq \sigma_{\text{critical}}\,,$$

where $\sigma_{\text{allowable}}$ is the allowable normal stress, σ_{critical} is the critical buckling stress of the truss member. The two-bar truss problem is formulated as a bicriteria optimization problem

$$\min\ f_1(x_1, x_2) = V\,, \quad f_2(x_1, x_2) = \sigma$$

$$\text{s.t.} \quad g_1(x_1, x_2) = \frac{\sigma_{\text{critical}}}{\sigma} - 1 \leq 0\,,$$

$$g_2(x_1, x_2) = \frac{\sigma}{\sigma_{\text{allowable}}} - 1 \leq 0\,,$$

$$1 \leq x_1 \leq 100\,, \quad 10 \leq x_2 \leq 1000\,, \tag{9}$$

The constant data used in the two-bar truss problem are shown in Table 1.

Table 1 Data for the two-bar truss problem

External Force	$F = 150$ KN
Normal stress limit	$[\sigma_{\text{allowable}}] = 400\ \text{N/mm}^2$
Thickness of the cross–section	$T = R - r = 2.5$ mm
Half width of the structure	$B = 750$ mm
Elastic modulus	$E = 210\,000\ \text{N/mm}^2$
Upper and lower bound of x_1	$1 \leq x_1 \leq 100$ mm
Upper and lower bound of x_2	$10 \leq x_2 \leq 1000$ mm

There are two design variables, two objective functions, and two constraints in this problem. The objective functions and the constraints are nonlinear. Before applying the proposed decision making procedure to this example, a detailed analysis of the problem is carried out in the next section. The results of this analysis are used to verify the output of the decision making procedure, and hence to validate the correctness of the proposed decision making procedure.

3.1.1 Analysis of the two-bar truss problem

Since there are only two design variables in this problem, the feasible regions in the design space and objective space can be presented graphically. They are shown in Fig. 3a and b, respectively. Figure 3a shows that the feasible region in the design space is not convex. Using the detection technique introduced by Fadel *et al.* (2002), we solve the problem using the Tchebycheff method and the weighting method at the two end points of the Pareto curve and at the equal weight point. The results of the process are listed in Table 2. The objective function values at the equal weight point are almost the same. This means that the Pareto curve at the equal weight point and in its neighbourhood is convex (in the objective space). The Tchebycheff method is used to generate the complete Pareto set. To verify the results of the Tchebycheff method (Tind and Wiecek 1997), the ε-constranint method and the weighting method are also used independently. The Pareto curves obtained by the three methods are almost identical. The results of all three methods are also plotted in Fig. 3a and b.

Figure 3b shows that the Pareto curve of the two-bar truss problem seems convex in the objective space although the feasible region in the design space is nonconvex and some of the Pareto points fall on a nonconvex constraint. This also verifies the results of the detection process. Figure 3a also shows that the super-objective function achieves some of its optimal points inside the feasible region in the design space. Table 3 lists the results from the weighting method. The ε-constraint method and

Table 2 Detection of the convexity of the two-bar truss problem

Tchebycheff method	w_1	w_2	x_1	x_2	f_1	f_2
1	0.00	1.00	41.53950	999.20378	0.8152107×10^6	0.28743834×10^3
2	0.50	0.50	39.17829	792.01668	0.67127526×10^6	0.33568100×10^3
3	1.00	0.00	33.87399	745.03459	0.56250306×10^6	0.40000665×10^3

Weighting method	w_1	w_2	x_1	x_2	f_1	f_2
1	0.00	1.00	41.54967	1000.0000	0.81582589×10^6	0.28728556×10^3
2	0.50	0.50	39.28917	803.10857	0.67816173×10^6	0.33255607×10^3
3	1.00	0.00	33.77161	749.58723	0.56250767×10^6	0.39999461×10^3

Table 3 Optimization results of the two-bar truss problem (by the weighting method)

w_1	w_2	x_1	x_2	f_1	f_2
0.00	1.00	41.54967	1000.00000	815 825.8887662	287.2855633
0.05	0.95	41.54967	1000.00000	815 825.8887210	287.2855633
0.10	0.90	41.54967	1000.00000	815 825.8886665	287.2855633
0.15	0.85	41.54967	1000.00000	815 825.8503783	287.2855768
0.20	0.80	41.54967	1000.00000	815 825.8502940	287.2855768
0.25	0.75	41.54967	1000.00000	815 825.8501842	287.2855769
0.30	0.70	41.54967	1000.00000	815 825.8500357	287.2855769
0.35	0.65	41.54873	999.92559	815 768.4712083	287.2997982
0.40	0.60	40.74618	935.26429	767 305.5039425	300.4080951
0.45	0.55	39.93229	864.67003	717 968.4032924	316.5613972
0.50	0.50	39.28917	803.10857	678 161.7286385	332.5560718
0.55	0.45	38.54059	752.14324	643 035.6335318	349.9042733
0.60	0.40	34.86203	749.78767	580 747.5511613	387.4316966
0.65	0.35	33.77163	749.58660	562 507.6839597	399.9945967
0.70	0.30	33.77162	749.58692	562 507.6759235	399.9946023
0.75	0.25	33.77162	749.58705	562 507.6727267	399.9946046
0.80	0.20	33.77162	749.58712	562 507.6710112	399.9946058
0.85	0.15	33.77162	749.58717	562 507.6699404	399.9946065
0.90	0.10	33.77162	749.58719	562 507.6692088	399.9946070
0.95	0.05	33.77162	749.58722	562 507.6686772	399.9946074
1.00	0.00	33.77161	749.58723	562 507.6682733	399.9946077

Table 4 Points used to construct the hyper-ellipse for the 2-bar problem

	w_1	w_2	x_1	x_2	f_1	f_2
1	0.00	1.00	41.53950	999.20378	815 210.6799793	287.4383361
2	0.50	0.50	39.17830	792.01668	671 275.2554480	335.6810005
3	1.00	0.00	33.87399	745.03459	562 503.0589894	400.0066494
				$\nu = 1.2350$		

the Tchebycheff method are used to verify the results of the weighting method. These results are listed in the Table 13.

The next step approximates the Pareto set obtained above using the hyper-ellipse. In order to generate more points to verify the approximation of the hyper-ellipse, the Tchebycheff method is used. Table 4 shows the points used to generate the hyper-ellipse and the parameter of the hyper-ellipse. Figure 4 shows the Pareto set of the two-bar truss problem and its approximation. Table 5

Table 5 Results of the approximation using hyper-ellipse

	w_1	f_1^*	w_2	f_2^*	f_2	$(f_2^*- f_2)/f_2^*$
1	0.00	815 210.68	1.00	287.43834	287.43834	0.00000000
2	0.05	794 697.43	0.95	292.72267	291.55652	0.00398380
3	0.10	776 169.69	0.90	297.84850	296.60606	0.00417139
4	0.15	759 552.82	0.85	302.75730	301.72564	0.00340753
5	0.20	744 342.80	0.80	307.51198	306.80613	0.00229537
6	0.25	701 673.99	0.75	322.72476	322.79524	− 0.00021838
7	0.30	717 303.81	0.70	316.80347	316.65408	0.00047157
8	0.35	705 004.36	0.65	321.42537	321.45953	− 0.00010627
9	0.40	693 350.88	0.60	326.09740	326.19802	− 0.00030856
10	0.45	682 180.33	0.55	330.85132	330.91195	− 0.00018326
11	0.50	671 275.26	0.50	335.68100	335.68100	0.00000000
12	0.55	660 998.62	0.45	340.59972	340.33320	0.00078249
13	0.60	650 407.11	0.40	345.95156	345.29820	0.00188859
14	0.65	640 177.33	0.35	351.46631	350.27061	0.00340206
15	0.70	629 866.65	0.30	357.21861	355.47602	0.00487822
16	0.75	619 413.82	0.25	363.24668	360.97706	0.00624815
17	0.80	608 759.61	0.20	369.60639	366.85474	0.00744479
18	0.85	597 834.49	0.15	376.36088	373.22999	0.00831883
19	0.90	586 556.49	0.10	383.59735	380.30312	0.00858772
20	0.95	573 144.37	0.05	397.12185	389.77231	0.01850701
21	1.00	562 503.06	0.00	400.00665	400.00665	0.00000000

lists the results of the approximation and its relative error at different points.

Note that the maximum error in the above table is due to numerical error in computation. The point that has the greatest error (which can be clearly seen in Fig. 3b is a runaway point in the Tchebycheff method[1]. Apart from this point, the greatest relative error is 0.859%. This scale of error is well within acceptable levels in engineering applications.

Having analysed the two-bar truss problem in detail, the proposed decision making procedure is applied to this problem in the next section.

3.1.2 Application of the decision making procedure

Following the steps of the proposed procedure, the two-bar problem can be analysed as follows.

1. Formulate the two-bar problem as a bicriteria optimization problem.
   The problem has been formulated earlier (9).
2. Set $w_1 = 1$ and $w_1 = 0$ to obtain the ideal points for objectives f_1 and f_2 respectively.
   Obtain the value of $f_1^{\min}$, $f_1^{\max}$,$f_2^{\min}$, and$f_2^{\max}$
3. Calculate the third point. Use equal weights to obtain the third point needed to construct the hyper-ellipse. Set $w_1 = w_2 = 0.5$ and solve (3) for f_1^* and f_2^* The values for the three points are listed in Table 5.

[1] Note that the weights in the weighting method can only generate distinct Pareto points in the interval [0.35,0.65]

4. Construct the hyper-ellipse.
   Using the three points obtained in the previous step solve the hyper-ellipse equation (4) for the value of the exponent ν. For the two-bar truss problem, $\nu = 1.235$. The approximation of the Pareto set can be expressed as

$$\left(\frac{f_1 - 815\,210.68}{562\,503.06 - 815\,210.68}\right)^{1.235} + \left(\frac{f_2 - 400.00}{287.44 - 400.00}\right)^{1.235} = 1\,. \tag{10}$$

5. Present the approximation of the Pareto set to the decision-maker, and obtain the preferred objective function value(s).
6. Depending on the number of values obtained, proceed to step a or b.
   (a) If the engineer (decision-maker) wishes to fix one preferred objective function value, the ε-constrained method can be used to formulate (1) as a single objective optimization problem.
   Assuming the engineer wants to limits the total volume to 700 000 mm^3, the normal stress σ will be about 320 N/mm^2 according to the approximated Pareto curve (the hyper-ellipse). To evaluate the design variables, set $f_1 = 700\,000$, and using the ε-constraint method (5), the design variables are obtained

$$x_1 = 39.6398\,, \quad x_2 = 837.464\,,$$
$$f_1 = 699\,999.99\,, \quad f_2 = 323.38\,.$$

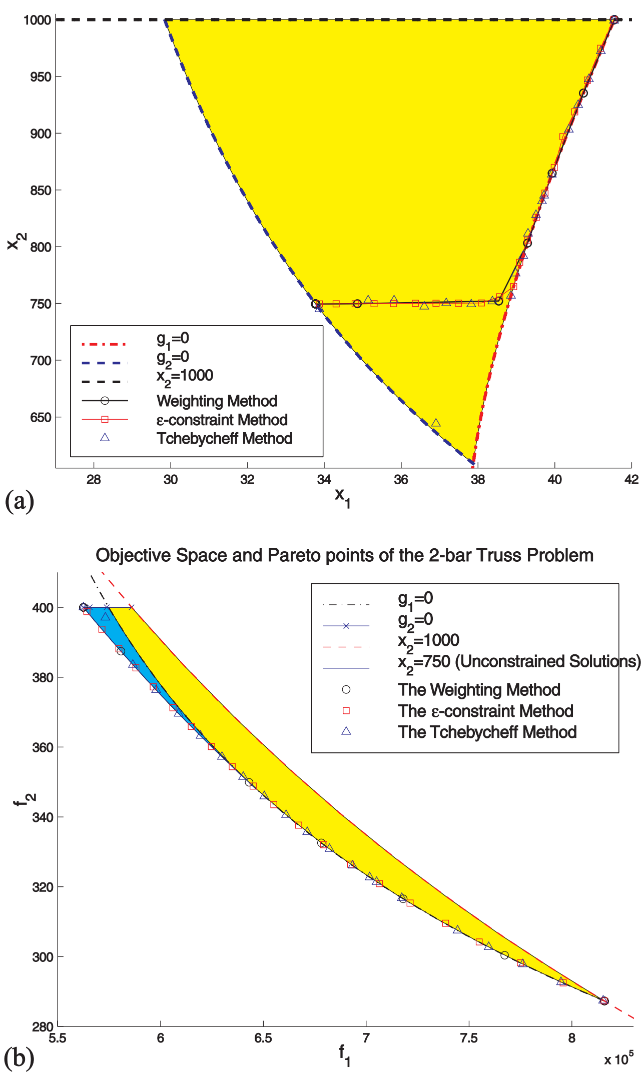

Fig. 3 Pareto set of the two-bar truss problem. (a) Pareto points in the design space, (b) Pareto points in the objective space

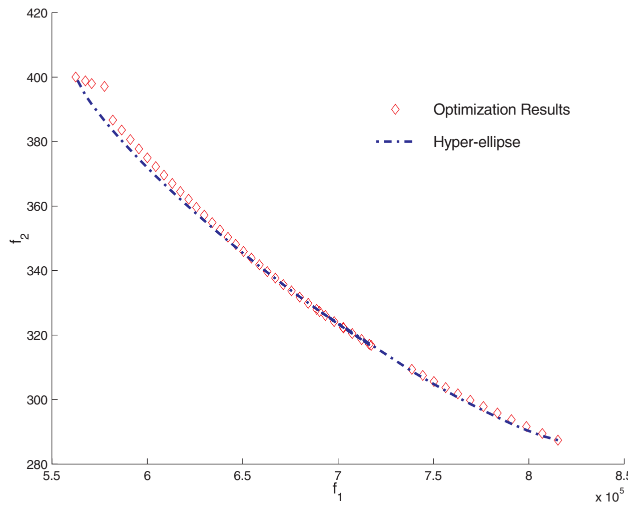

Fig. 4 The approximation of the two-bar truss problem

The solution of this problem is the preferred solution for the BCOP and hence, is the solution of the two-bar truss problem.

(b) If the engineer wishes to target two objective function values, the goal programming method can be used to obtain the values of the design variables and those of the deviation variables at the preferred point.
Assuming the engineer wants the volume of the structure to be around (less than) 650 000 mm^3 and the normal stress around 350 N/mm^2, a goal programming formulation can be set. Let the goals be $b_1 = 650\,000$ and $b_2 = 350$, (6) yields

$$x_1 = 38.1824\,,\; x_2 = 766.148\,,\; f_1 = 643{,}034.82\,,$$
$$f_2 = 349.98\,.$$

Because both of the objective function values are specified, generally it is not possible to achieve both goals simultaneously. The goal programming method tries to achieve both goals, and the final results have an under- or an over-achievement. Since the objective functions are normalized inside the program, the under- or over-achievement is a relative measure, which indicates how far the final results are from the goals. The goals achieved here are not exactly what the DM specified, they have 0.027 and 0.000068 under-achievement for objective 1 and objective 2 respectively.

7. Assuming the decision-maker is satisfied with the results, the decision-making process ends.
As mentioned before, although the optimization procedure requires that the objective functions be normalized, the approximated Pareto curve presented to the DM must be in its original scale. The normalized data may have no meaning to the DM. After the preference information is obtained, it must be normalized so that it can be used in the final optimization process.

3.2 Three-bar truss problem

According to Koski (1985), the three-bar truss is a nonconvex example of the bicriteria optimization problem

$$\min\; f_1(\underline{x}) = \sum_{i=1}^{3} x_i\,, \quad f_2(\underline{x}) = 0.75\,\delta_x + 0.25\,\delta_y\,,$$

$$\text{s.t.}\quad g_i(\underline{x}) = \frac{N_i}{x_i} - [\sigma_i] \le 0, \quad i = 1, \ldots, 3\,,$$

$$10 \le x_i \le 200\,, \quad i = 1, \ldots, 3\,, \tag{11}$$

Figure 5 shows the structure of the three-bar truss problem.

Here x_i is the cross-sectional area of the truss member i which serves as design variable; $\underline{x} = (x_1, x_2, x_3)$; δx and δy are the horizontal and vertical displacements of the node 1 in Fig. 5; N_i is the internal force of the i-th truss member.

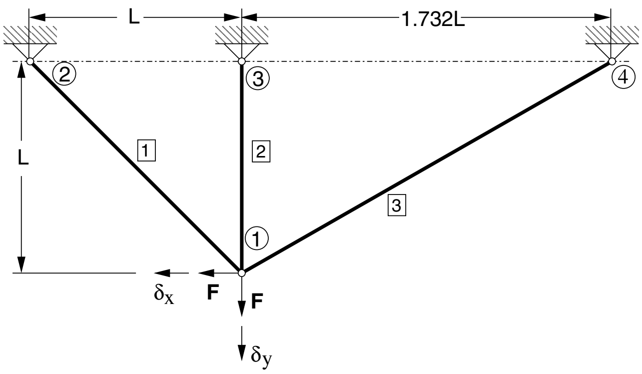

Fig. 5 The three-bar truss problem

Since the three-bar truss problem is a statically indeterminate problem, N_i can be solved using a finite element package or an analytical method; $[\sigma_i]$ is the maximal allowable normal stress of the i-th truss member; $[\sigma_i]$ is a constant.

All the constant values used to solve the problem are listed in Table 6. Objective function $f_1(x)$ is the volume of the structure. Objective function $f_2(x)$ is an arbitrary weighted sum of the displacements of node 1 selected by the designer.

Constraint $g_i(x)$ is the stress constraint of the i-th truss member.

Table 6 Data for the three-bar truss problem

External Force	F	= 20 KN
Normal stress limit	$[\sigma_{\max}]$	= 200 N/mm^2
Height of the structure	L	= 1000 mm
Elastic modulus	E	= 200 000 N/mm^2
Lower bound of x_i	x_i	= 10 mm^2
Upper bound of x_i	x_i	= 200 mm^2

3.2.1 Analysis of the three-bar problem

Since there are three design variables in this problem, it is hard to display the feasible region in the design space graphically. The detection technique is used to probe the convexity of the Pareto curve at the equal weight point. Different methods are used to solve this problem in order to verify the results. Figure 6 shows the results obtained by different methods.

Figure 6 shows that the Pareto set of the three-bar truss problem is a nonconvex set. Also this set consists of two disconnected parts when using the discretization selected. The best approximation of this kind of Pareto set is to approximate each part of the Pareto set separately. To do this, much more information is needed. In real engineering problems, the luxury of having the real Pareto set is not affordable. Knowing that the Pareto set is nonconvex is considered a big advantage. In this case, one hyper-ellipse is used to approximate the entire Pareto set. As said in the decision making procedure section, the approximation error will be eliminated after the final optimization run.

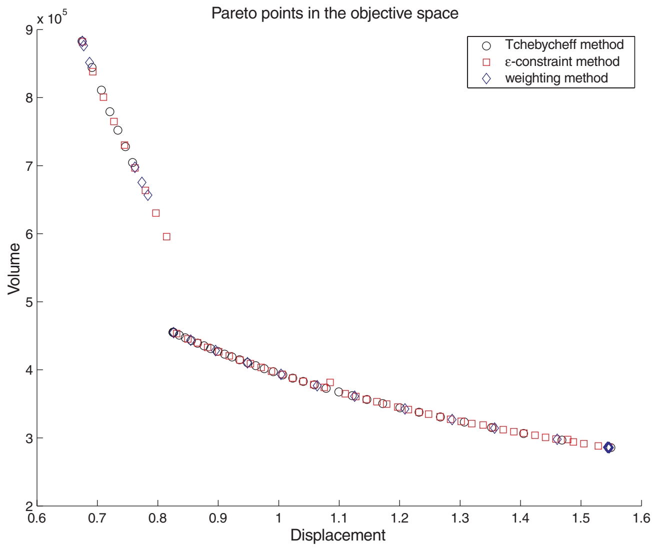

Fig. 6 Pareto points of the three-bar truss problem

Table 7 Pareto points used to construct the hyper-ellipse

w_1	w_2	x_1	x_2	x_3	f_1	f_2
Tchebycheff method						
0.00	1.00	199.998	200.000	200.000	882 839.3266	0.6746
0.50	0.50	10.000	41.450	187.836	431 264.7500	0.8875
1.00	0.00	10.000	46.680	112.496	285 814.0504	1.5494
$\nu = 2.4844$						
ε-constraint method						
0.00	1.00	199.646	200.000	200.000	881 942.3255	0.6753
		10.000	43.097	153.730	365 006.9594	1.1100
1.00	0.00	10.000	46.705	112.641	286 353.7980	1.5458
$\nu = 2.0143$						
weighting method						
0.00	1.00	200.000	200.000	200.000	882 442.7076	0.6751
0.50	0.50	10.000	40.965	200.000	454 706.8598	0.8262
1.00	0.00	10.000	46.362	112.454	285 637.2426	1.5479
$\nu = 2.7219$						

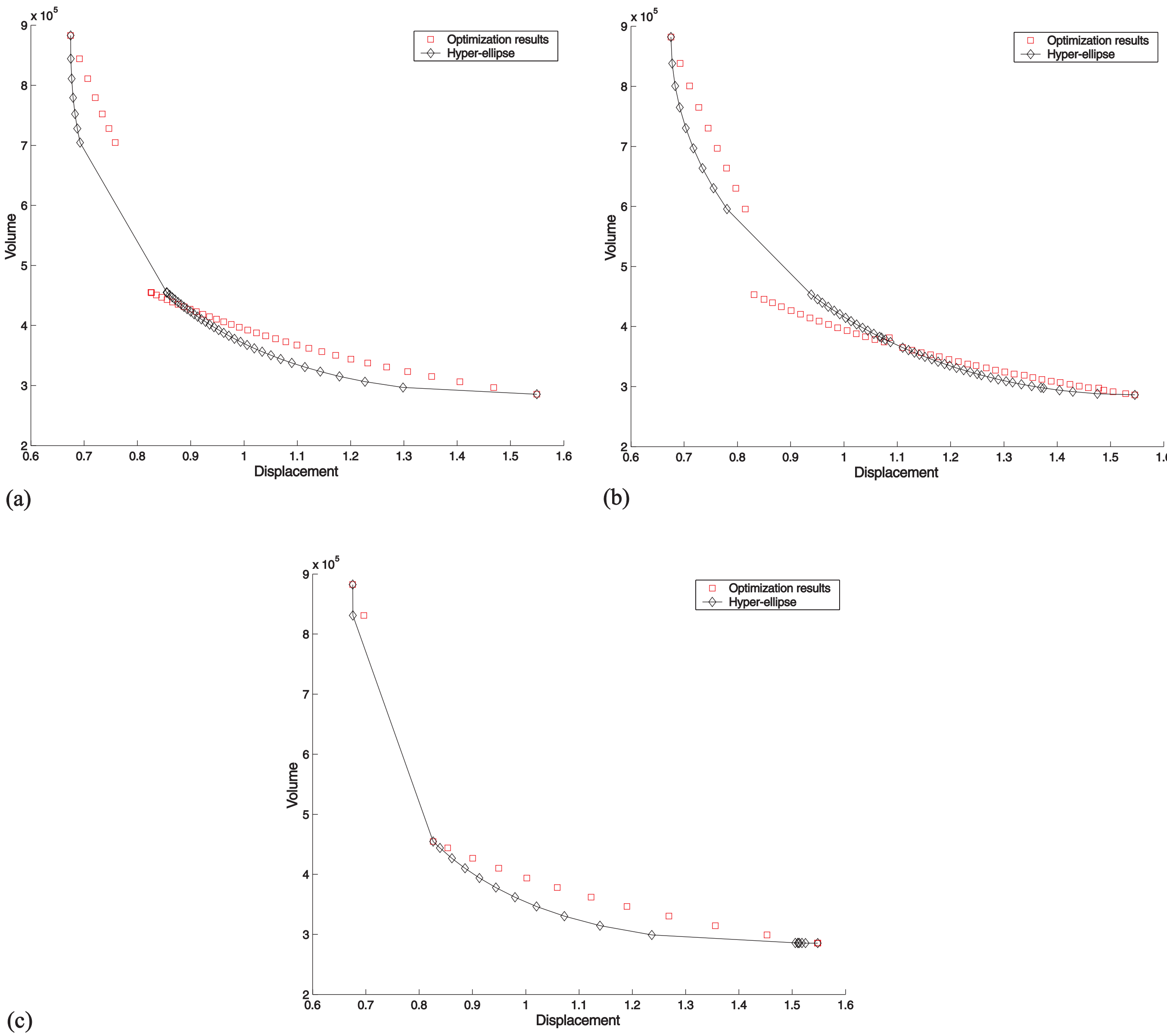

Fig. 7 a–c Approximation of the Pareto set of the three-bar truss problem

To approximate the Pareto set of this problem, the extreme points of the Pareto set need to be calculated. These are obtained by solving each objective in turn as a single objective optimization problem. The equal weight point is used to generate the third point. Using the three points in the objective space, the hyper-ellipse is constructed. The results are listed in Table 7. Figure 7 shows the Pareto set and its approximation in the objective space.

3.3
Application of the decision making procedure

Applying the decision-making procedure to the three-bar truss problem, the following steps are carried out.

1. Formulate the three-bar truss problem as a bicriteria optimization problem (11).
2. Obtain the extreme points of the Pareto. The results are listed in Table 8.
3. Calculate the third point.
   Since the Pareto set is a nonconvex set, the ε-constraint method is used to solve this problem. Because there is no extra information to help select the third point, a point where f_z is the middle of the range used to obtain the third point (Table 7).
4. Construct the hyper-ellipse.
   Using results from Table 7, solve the hyper-ellipse equation (4) for the value of the exponent ν. Using the data obtained by the ε-constraint method, $\nu = 2.0143$. The hyper-ellipse, which is the approximation of the Pareto set, can be expressed as

$$\left(\frac{f_1-285\,814.04}{882\,839.33-285\,814.04}\right)^{2.0143}+$$

$$\left(\frac{f_2-0.67}{1.55-0.67}\right)^{2.0143}=1\,. \tag{12}$$

5. Present the approximation of the Pareto set to the decision-maker.
6. Obtain the preferred objective function value.
7. Depending on the number of values obtained, proceed to step a or b.
   (a) If one objective function value is specified, the ε-constrained method (5) is used to formulate the problem as a single objective optimization problem.
   Assuming that the designer wants to limit the combined displacement at node 1 to 0.9 mm after balancing between the volume (weight) and the displacement. Through the hyper-ellipse equation, the volume is around $5\times10^5\,\text{mm}^3$.
   Using the ε-constraint method, set $\varepsilon_2=0.9$ and solve the following single objective optimization problem:

$$\min\ f_1(\underline{x})=\sum_{i=1}^{3}x_i\,,$$

$$\text{s.t.}\quad f_2(\underline{x})=0.75\,\delta_x+0.25\,\delta_y\le\varepsilon_2\,,$$

$$g_i(\underline{x})=\frac{N_i}{x_i}-[\sigma_i]\le 0\,,\quad i=1,\ldots,3\,,$$

$$10\le x_i\le 200\,,\quad i=1,\ldots,3\,, \tag{13}$$

   Solving the above problem, we obtain

$$x_1=10.0000\,,\ x_2=41.4545\,,\ x_3=187.175\,,$$

$$f_1=430\,320.2412\,,\ f_2=0.9000\,.$$

   Since the approximation for this problem has relatively big errors, the DM may not be satisfied with this solution. The volume could be far from the allowed value. So the DM re-specifies the allowable displacement at node 1 to be 0.83 mm.
   Set $\varepsilon_2=0.83$, solve (13)

$$x_1=10.0000\ x_2=41.2174\ x_3=198.9340\,,$$

$$f_1=453\,624.4606\,,\ f_2=0.8300\,.$$

   This is the preferred solution of the designer (DM).
   (b) If the decision-maker sets two objective function values, the goal programming method is used to obtain the values of the design variables and those of the deviation variables at the preferred location. Assuming the engineer wants both the volume of the structure at $4.5\times10^5\,\text{mm}^3$ and the displacement at 8.5 mm, a goal programming formulation (6) can be set with the goals $b_1=4.5\times10^5$ and $b_2=8.5$ and yields

$$x_1=17.9401\,,\ x_2=40.2693\,,\ x_3=195.9840\,,$$

$$f_1=457\,607.55\,,\ f_2=0.8504\,.$$

   The results from goal programming display some over-achievement for the two objectives. This can be seen from the figure of the approximating curve (Fig. 7b), part of the actual Pareto set is at the left side of the hyper-ellipse. The points on this part yield a better solution than the points on the hyper-ellipse.

3.4 Ten-bar truss problem

The ten-bar truss problem described by Haftka and Gurdal (1993) is another commonly used example in optimization research. The common use of the example is to use the areas of the ten truss members as design variables, and as the objective, to minimize the total weight (or volume). Figure 8 shows the structure of this problem. Equation (14) gives the formulation of the commonly used single objective optimization formulation,

$$\min\ \sum_{i=1}^{10}x_i\,,$$

$$\text{s.t.}\quad g_i(\underline{x})=\frac{N_i}{x_i}-[\sigma_i]\le 0\,,\quad i=1\,,\ldots,10\,,$$

$$0.1\le x_i\le 30\,,\quad i=1,\ldots,10\,, \tag{14}$$

where x_i are the i-th bar's cross–sectional areas. They serve as design variables; N_i are the internal forces of the i-th truss members. The problem is a second–order statically indeterminate problem; N_i can be solved numerically by the finite element method or analytically by solving the statically indeterminate problem; $[\sigma_i]$ is the maximum allowable stress without failure.

Because the ten-bar truss problem is a classical optimization problem, it is meaningful to convert it to a multiple criteria optimization problem. The first step is to convert it into a bicriteria optimization problem. The first objective remains the weight (volume) of the structure, which needs to be minimized. The second objective, a classical constraint, is to minimize the vertical displacement at node 2 (see Fig. 8). Minimizing the total volume of the structure results in decreasing the cross-sectional areas of each truss member (or of some of them). These cross-sectional areas are the design variables of this problem. To minimize the displacement, the stiffness of the

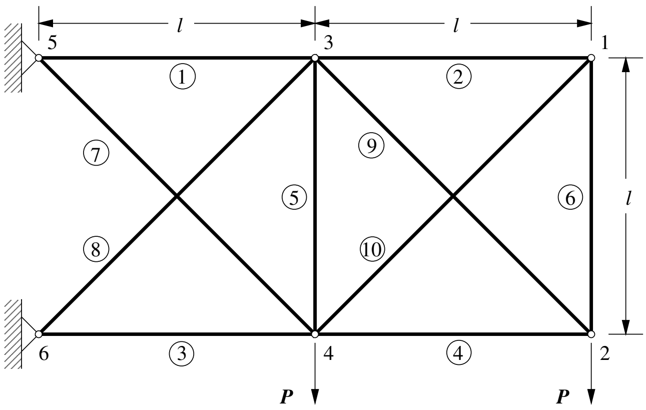

Fig. 8 The ten-bar truss problem

structure must be increased. This means that the cross-sectional areas must be increased because the topology of the structure is predetermined. So the two objectives conflict with each other, and the bicriteria optimization problem can be formulated as follows:

$$\min\ f_1(\underline{x}) = \sum_{i=1}^{10} x_i\,, \quad f_2(\underline{x}) = \delta_5\,,$$

$$\text{s.t.} \quad g_i(\underline{x}) = \frac{N_i}{x_i} - [\sigma_i] \le 0\,, \quad i = 1, \ldots, 10\,,$$

$$0.1 \le x_i \le 30\,, \quad i = 1, \ldots, 10\,, \tag{15}$$

where δ_5 is the vertical displacement at node 2 in Fig. 8, δ_5 is a complex function of the design variables x; x and N_i are the same as in (14); N_i can be solved by the finite element method or by the analytical method. Table 8 shows the data used to solve this problem.

Table 8 Data for the ten-bar truss problem

External Force	P	= 100 Kips
Normal stress limit	$[\sigma_{\max}]$	= 25 ksi
Height of the structure	l	= 360 in
Elastic modulus	E	= 30 000 ksi
Lower and upper bounds of x_i	0.1 ≤	x_i ≤30 in
initial point	x_i	= 5.0 in

3.4.1 Analysis of the ten-bar truss problem

The first objective function (the total volume of the truss structure) in (15) is a linear function of the design variables. The second objective function (the displacement) is a nonlinear objective function of the design variables. The problem has ten design variables, this makes it hard to solve analytically by hand. On the other hand, it is not big enough to use commercial FEM packages, because the time needed to initialize the software package is much more than what is needed to solve the problem. A procedure was written by the authors to solve this problem analytically and to generate the FORTRAN code automatically. This procedure was written in MAPLE and is available from the authors.

Also because the ten-bar truss problem has ten design variables, it is impossible to graph the feasible region in the design space. To verify the results, different optimization techniques are used to solve the problem, including the weighting method, the ε-constraint method, and the Tchebycheff method.

3.4.2 The weighting method

Using the weighting method, the bicriteria ten-bar truss problem becomes

$$\min\ w_1 f_1(\underline{x}) + w_2 f_2\,,$$

$$\text{s.t.} \quad g_i(\underline{x}) = \frac{N_i}{x_i} - [\sigma_i] \le 0\,, \quad i = 1, \ldots, 10\,,$$

$$x_i \ge 0\,, \quad i = 1, \ldots, 10\,,$$

$$w_1 + w_2 = 1\,, \quad w_i \ge 0\,, \quad i = 1, 2\,, \tag{16}$$

where $f_1(\underline{x})$ and $f_2(\underline{x})$ are the objective functions, as defined in (15); w_1 and w_2 are the weights.

All other variables are as described in (15).

The optimization program DOT (Design Optimization Tool Vanderplaats, Miura & Associates 1993) is used to optimize the ten-bar truss problem. In order to get the Pareto set of the problem by varying the weights associated with each objective, the objective functions need to be normalized. This is very important in this problem, because the order of magnitude of the total volume is 10^6 while that of the displacement is less than 10. The natural normalization method presented by Lieberman (1991) is used in this approach.

The design variables, the objective function values as well as the corresponding weights are listed in Table 9.

Using a step size of 1/20 to change the weight in (16), the weighting method yields a set of Pareto points, which are listed in Table 10.

In Table 10, the data shows that at the end of the Pareto curve where f_1 approaches its minimum, the derivative of the curve does not approach infinity. The function values change little when the weights change. The indifference line of the weighting method ($w_1 f_1 + w_2 f_2 = C$) rotates around the end point of the Pareto curve. This can be seen in Fig. 9. In order to avoid computational difficulties, the weighted sum of the two normalized objectives is multiplied by 100. The numerical difficulties are discussed in Sect. 4.

Table 9 Extreme points of the ten-bar truss problem

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
1	30.00	30.00	30.00	30.00	0.10	30.00	30.00	30.00	30.00	30.00
2	7.94	0.10	8.06	3.95	0.10	0.10	5.74	5.57	5.57	0.10

	$f_{\min}$	$f_{\max}$
1	15 936.56	115 114.75
2	1.303	7.197

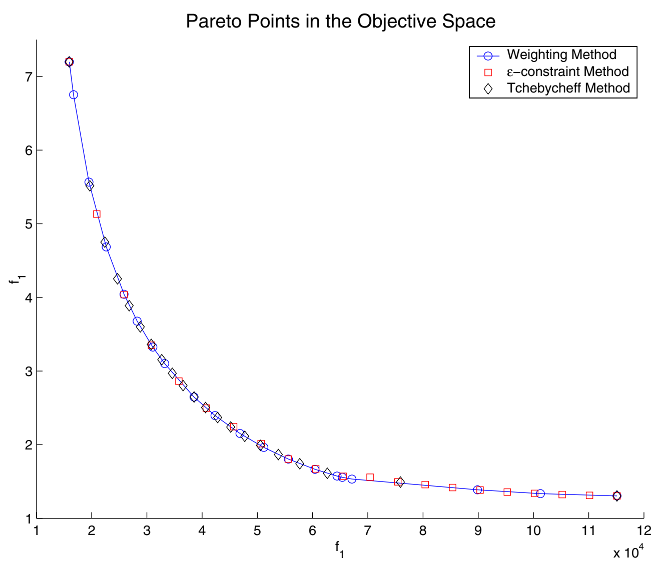

Fig. 9 The approximation of the ten-bar truss problem

Table 10 Results obtained by the weighting method

	w_1	w_2	f_1	f_2
1	0.00	1.00	115 114.75238804	1.30340166
2	0.05	0.95	101 267.03121038	1.33408506
3	0.10	0.90	89 860.30947274	1.38624019
4	0.15	0.85	67 121.73043662	1.53263072
5	0.20	0.80	65 388.58623101	1.55553372
6	0.25	0.75	64 432.38994589	1.57232872
7	0.30	0.70	60 431.79083217	1.66581196
8	0.35	0.65	55 615.28192718	1.80353889
9	0.40	0.60	51 166.82895283	1.96176915
10	0.45	0.55	46 860.45841514	2.15229851
11	0.50	0.50	42 346.20767985	2.39515314
12	0.55	0.45	38 521.28678384	2.64739800
13	0.60	0.40	33 212.90573230	3.10031070
14	0.65	0.35	31 058.18066569	3.32422562
15	0.70	0.30	28 224.93416848	3.67805761
16	0.75	0.25	25 859.35577662	4.04043420
17	0.80	0.20	22 642.28442690	4.68492560
18	0.85	0.15	19 506.17643098	5.56498305
19	0.90	0.10	16 709.08192602	6.75024377
20	0.95	0.05	15 937.43385672	7.19650367
21	1.00	0.00	15 936.56260077	7.19692994

To verify the results obtainedby the weighting method, the Tchebycheff and e-constraint methods are used to solve this problem. The results are listed in Tables 15 and 16 and in Fig. 9.

From the data in Table 10 and Fig. 10, it seems clear that the ten-bar truss problem has a convex Pareto curve. It is mentioned in many citations (e.g. Haftka and Gurdal 1993) that most of the structural optimization problems are nonconvex, but since the results of the three methods show near identical points, the problem can be assumed to be convex (Fadel *et al.* 2002).

After analysing the Pareto set of the ten-bar truss problem, the next step is to approximate it using the hyper-ellipse to further validate the proposed methodology.

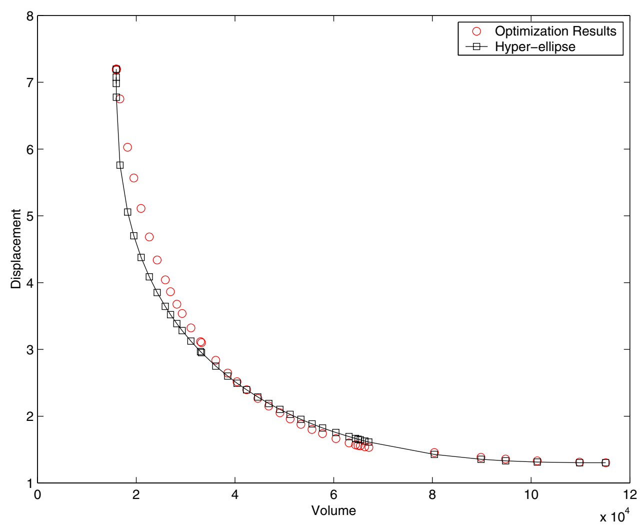

Fig. 10 Approximation of the Pareto curve of the ten-bar truss

4 Approximation of the Pareto curve of the ten-bar problem

To approximate the Pareto set of the ten-bar truss problem, three points are needed to construct the hyper-ellipse. The two extreme points of the Pareto set are obtained in the normalization process, the third point

Table 11 Parameters of the hyper-ellipse to approximate the Pareto curve of the ten-bar truss problem

	w_1	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
1	1.0	7.94	0.10	8.06	3.95	0.10	0.10	5.74	5.57	5.57	0.10
2	0.0	30.00	30.00	30.00	30.00	0.10	30.00	30.00	30.00	30.00	30.00
3	0.5	24.80	0.10	19.39	12.89	0.10	0.10	6.92	17.21	18.37	0.10

	$f^{\min}$	$f^{\max}$	$f^{3\text{rd}}$
1	15 936.56260077	115 114.75238804	42 346.20767985
2	1.30340166	7.19692994	2.39515314
		$\nu = 2.7342$	

Table 12 Results of the approximation of ten-bar truss problem

	w_1	f_1^*	w_2	f_2^*	f_2	relative error
1	0.00	115 114.8	1.00	1.303402	1.303402	0.0000000
2	0.05	101 267.0	0.95	1.334085	1.313318	0.0155662
3	0.10	89 860.31	0.90	1.386240	1.354989	0.0225439
4	0.15	67 121.73	0.85	1.532631	1.613612	−0.0528380
5	0.20	65 388.59	0.80	1.555534	1.646924	−0.0587515
6	0.25	64 432.39	0.75	1.572329	1.666346	−0.0597952
7	0.30	60 431.79	0.70	1.665812	1.756248	−0.0542896
8	0.35	55 615.28	0.65	1.803539	1.885101	−0.0452233
9	0.40	51 166.83	0.60	1.961769	2.027505	−0.0335086
10	0.45	46 860.46	0.55	2.152298	2.190924	−0.0179463
11	0.50	42 346.21	0.50	2.395153	2.395153	−0.0000000
12	0.55	38 521.29	0.45	2.647398	2.601004	0.0175246
13	0.60	33 212.91	0.40	3.100311	2.953072	0.0474916
14	0.65	31 058.18	0.35	3.324226	3.125166	0.0598815
15	0.70	28 224.94	0.30	3.678057	3.386518	0.0792645
16	0.75	25 859.35	0.25	4.040436	3.645238	0.0978106
17	0.80	22 642.29	0.20	4.684925	4.086638	0.1277049
18	0.85	19 506.18	0.15	5.564983	4.701800	0.1551097
19	0.90	16 708.75	0.10	6.750439	5.758702	0.1469146
20	0.95	15 937.43	0.05	7.196504	7.076471	0.0166793
21	1.00	15 936.56	0.00	7.196930	7.196930	0.0000000

can be calculated using the equal weight point. Let $w_1 = w_2 = 0.5$, solving (3), the third point is obtained. Using the three points to solve the hyper-ellipse equation (4), the exponent ν can be calculated. All three points used to construct the hyper-ellipse and the exponent are listed in Table 11.

Table 12 contains the optimization results yielded by the weighting method, the data obtained from the hyper-ellipse, and the relative error of the approximation. Figure 10 shows that the slope of the Pareto curve of the ten-bar truss problem at the end $f_2 = f_2^{\min}$ is near 0. The accuracy of the approximation at this end is very satisfactory.

At the other end of the Pareto curve, the actual slope is not infinity. Because the slope of the hyper-ellipse is infinity at this end, the approximation introduces some error. Numerical experiments show that using a different middle point does not lead to significant improvement in the accuracy of the approximation. This error, however, will not affect the value of the approximation since in the decision making process, there is a final optimization run to obtain the values of the design variables and the objective functions.

The analysis of the Pareto curve of the ten-bar truss problem is again used to verify the results of the proposed decision making procedure. For a common engineering application, the detailed analysis is unnecessary, and generally, impossible. In the next section, the proposed decision making procedure is demonstrated by applying it to the ten-bar truss problem.

4.0.3
Application of the decision making procedure

Again, following the steps of the proposed decision making procedure, the ten-bar truss problem can be solved as follows.

1. Formulate the ten-bar truss problem as a bicriteria optimization problem (BCOP)
2. Set $w_1 = 1$ and $w_1 = 0$ to obtain the ideal points for objective f_1 and f_2 respectively.
   Calculate the values of $f_1^{\min}$, $f_1^{\max}$, $f_2^{\min}$, and $f_2^{\max}$, these values are listed in Table 11.
3. Calculate the third point.
   Since there is no extra information to help select the third point, equal weights are used for the third point needed to construct the hyper-ellipse. Set $w_1 = w_2 = 0.5$ and solve (3) for f_1^* and f_2^*. The values for the three points are also listed in Table 11.
4. Construct the hyper-ellipse.
   Using the three points obtained in the previous step, solve the hyper-ellipse equation (4) for the value of the exponent ν. For the ten-bar truss problem, $\nu = 2.7342$. The approximation of the Pareto set can be expressed as

$$\left(\frac{f_1 - 15\,936.56}{115\,114.75 - 15\,936.56}\right)^{2.7342} + \left(\frac{f_2 - 1.303}{7.197 - 1.303}\right)^{2.7342} = 1\,. \quad (17)$$

5. Present the approximation of the Pareto set to the decision-maker, and obtain the preferred objective function value(s).
6. Depending on the number of values obtained, proceed to step a or b.
   (a) If one objective function value is specified as the preferred solution, the ε-constrained method is used to formulate the problem as a single objective optimization problem.
   Assuming that the displacement is more critical in this structure; the engineer wants to limit the displacement at node 2 (Fig. 8) to less or equal to 2 in., according to the Pareto curve (Fig. 10), the total volume of the structure will be around $50\,000\,\text{in}^3$.
   Using the ε-constraint method, set $\delta_2 = 2$ and solve (5), the actual design variables and the objective functions are obtained
   $$x_1 = 29.9934, \quad x_2 = 0.10000$$
   $$x_3 = 22.7425, \quad x_4 = 15.5764$$
   $$x_5 = 0.10000, \quad x_6 = 0.10000$$
   $$x_7 = 7.44306, \quad x_8 = 20.7933$$
   $$x_9 = 21.8079, \quad x_1 = 0.10000$$
   $$f_1 = 50\,229.79, \quad f_2 = 2.0001.$$
   This is the solution according the preference of the engineer (DM).
   (b) If the decision-maker selects two objective function values, the goal programming method is used to obtain the values of the design variables and those of the deviation variables at the preferred location.
   Assuming the engineer wants both the volume (mass) of the structure at 58 000 lb and the displacement at 1.6 in, a goal programming formulation can be set with the goals (from Fig. 10) $b_1 = 58\,000$ and $b_2 = 1.60$. Solving the problem yields
   $$x_1 = 30.0000, \quad x_2 = 0.10000$$
   $$x_3 = 29.4505, \quad x_4 = 18.5412$$
   $$x_5 = 0.10108, \quad x_6 = 26.1066$$
   $$x_7 = 7.60888, \quad x_8 = 26.1066$$
   $$x_9 = 26.3262, \quad x_1 = 0.100000$$
   $$f_1 = 58\,804.55, \quad f_2 = 1.71.$$

As stated before, because both values of the objectives are specified, it is not common that both goals can be reached exactly, however the results will be very close to those of the approximation. This solution is the optimal solution of the ten-bar truss problem according the engineer's preference.

The approach proposed is in our opinion more effective to designers than the usually applied a priori method. This is particularly clear in this example. In the a priori weighting method, the designer would have made an assumption that for instance volume or weight was twice as important as deflection. This would have resulted in a solution of 3.3". There is no relationship between the preference information and the results obtained. The engineer could have easily been satisfied with another weighting. Providing the trade-off curve not only conveys sensitivity type information, but also, real numerical trade-off values between the two objectives that can be used much more effectively. As a designer, one can also assess that there is no point in trying to reduce deflection to less than 1.4" since the volume or weight cost will increase very significantly for very little gain in stiffness.

5 Numerical difficulties

There are some numerical difficulties in solving the ten-bar truss problem. The problems are described here followed by the steps used to circumvent them.

5.1 The vertical displacement of node 2 is practically insensitive to some of the design variables

When the objective function is minimized, especially when the displacement of node 2 is minimized, the value of the displacement varies little when changing the weights and the initial points. The value of the volume, however, changes considerably. This causes problems when using different methods to solve this problem. For example, when setting $w_1 = 0$ and $w_2 = 1$ in the weighting method to minimize the displacement of point 2, the displacement obtained is almost the same as the result of the Tchebycheff method by setting $\delta_1 = 0$ and $\delta_2 = 1$. But the volumes of the structure obtained by these two methods are different. Using the approaches mentioned below can reduce this effect.

5.2 Convergence problem of the weighting method

In the optimization process, after normalization, the objective functions are between 0 and 1. Since the weights w_i are also normalized, $\sum w_i = 1$, the super-objective $w_i f_1 + w_2 f_2$ is less than 1. The optimizer has some difficulties converging, even using double precision for all variables. To avoid this problem, the super-objective function is multiplied by 100.

5.3 For the Tchebycheff method, the constraints associated with β are not satisfied with the original formulation

The reason for this is that the values of the constraints are too small. They do not exert enough penalty to

Table 13 Optimization results of the two-bar truss problem (the ε-constraint method)

w_1	w_2	x_1	x_2	f_1	f_2
0.00	1.00	41.53900	1000.00000	815 616.2647762	287.3593994
0.05	0.95	41.19213	974.67569	795 757.5061814	292.5119871
0.10	0.90	40.84982	946.69116	774 990.8361759	298.2356418
0.15	0.85	40.51940	918.88236	754 928.8322935	304.2086222
0.20	0.80	40.21212	896.98683	738 541.0787766	309.5466956
0.25	0.75	39.98593	869.70509	721 324.8029287	315.3522618
0.30	0.70	39.74218	847.30690	706 397.7066049	320.8904050
0.35	0.65	39.51527	825.75031	692 402.1651683	326.4610596
0.40	0.60	39.28765	805.75797	679 331.3449685	332.0599947
0.45	0.55	39.08577	786.24000	667 119.1275880	337.6466828
0.50	0.50	38.92909	764.88356	655 057.3850718	343.5476755
0.55	0.45	38.56364	755.91288	645 039.2056957	348.8267624
0.60	0.40	38.09279	750.38858	634 821.2529022	354.4305254
0.65	0.35	37.49084	750.27635	624 742.8959137	360.1481773
0.70	0.30	36.90636	750.16780	614 958.6606948	365.8782614
0.75	0.25	36.36953	750.06841	605 973.5196481	371.3033551
0.80	0.20	35.80049	749.96305	596 450.5671475	377.2315975
0.85	0.15	35.29462	749.86922	587 985.7401013	382.6623472
0.90	0.10	34.80149	749.77751	579 735.0223654	388.1083620
0.95	0.05	34.30724	749.68536	571 466.5783321	393.7238473
1.00	0.00	33.87121	749.60308	564 172.5756621	398.8141949

Table 14 Optimization results of the two-bar truss problem (the Tchebycheff method)

w_1	w_2	x_1	x_2	f_1	f_2
0.00	1.00	41.53950	999.20378	815 210.6799793	287.4383361
0.05	0.95	41.19935	972.33592	794 697.4251825	292.7226726
0.10	0.90	40.88772	947.60435	776 169.6889984	297.8485007
0.15	0.85	40.60825	924.88210	759 552.8206574	302.7572974
0.20	0.80	40.36523	903.12444	744 342.8019673	307.5119808
0.25	0.75	39.66687	840.03759	701 673.9931968	322.7247619
0.30	0.70	39.92148	863.67737	717 303.8088933	316.8034697
0.35	0.65	39.72091	845.13237	705 004.3638403	321.4253678
0.40	0.60	39.51258	827.91629	693 350.8808010	326.0974017
0.45	0.55	39.30074	811.55212	682 180.3335718	330.8513181
0.50	0.50	39.17830	792.01668	671 275.2554480	335.6810005
0.55	0.45	38.97851	776.52838	660 998.6148147	340.5997190
0.60	0.40	38.85975	756.87035	650 407.1092314	345.9515609
0.65	0.35	38.37393	751.96138	640 177.3332190	351.4663137
0.70	0.30	37.82235	749.32296	629 866.6517645	357.2186131
0.75	0.25	37.17255	750.21549	619 413.8168984	363.2466748
0.80	0.20	36.60370	747.32243	608 759.6065755	369.6063875
0.85	0.15	35.81685	752.75429	597 834.4861045	376.3608773
0.90	0.10	35.14107	752.75877	586 556.4943843	383.5973495
0.95	0.05	36.90597	644.16724	573 144.3681195	397.1218504
1.00	0.00	33.87399	745.03459	562 503.0589894	400.0066494

pull back the search directions. Using the same approach as in the weighting method, the values of the constraints are increased (multiplied by 100 for each objective, that is $g_i = 100\lambda_i\ f_i - \beta \leq 0$; the super-objective β is also increased ($\beta \times 1000$). After increasing the constraints and the super-objective, the optimizer converges smoothly.

A general note to the statically indeterminate truss problems used above. When selecting the displacement and total weight (volume) as objectives, the displacement is not as sensitive to some of the design variables cross sectional areas. Physically, to minimize the displacement, the stiffness needs to be increased. Since the geometry and the topology of the structure are fixed, the only way

Table 15 Results of the ten-bar truss problem using the weighted Tchebycheff method

	w_1	w_2	f_1	f_2
1	0.00	1.00	114 858.84054587	1.30372920
2	0.05	0.95	75 917.26217639	1.49070752
3	0.10	0.90	62 666.74536810	1.61173730
4	0.15	0.85	57 674.10552588	1.74095937
5	0.20	0.80	53 759.82908485	1.86525537
6	0.25	0.75	50 513.14095987	1.98835088
7	0.30	0.70	47 699.01969717	2.11249093
8	0.35	0.65	45 152.16458896	2.23865014
9	0.40	0.60	42 803.26907408	2.36807109
10	0.45	0.55	40 607.90168684	2.50373943
11	0.50	0.50	38 529.76726677	2.64655623
12	0.55	0.45	36 531.22645056	2.79985125
13	0.60	0.40	34 580.91596109	2.96756530
14	0.65	0.35	32 668.91770644	3.15094344
15	0.70	0.30	30 752.36466149	3.35891021
16	0.75	0.25	28 805.87919272	3.59922582
17	0.80	0.20	26 795.58887335	3.88647880
18	0.85	0.15	24 682.37567927	4.25112141
19	0.90	0.10	22 373.48100214	4.74992267
20	0.95	0.05	19 664.82323129	5.51973114
21	1.00	0.00	15 945.53208383	7.19134173

Table 16 Results of the ten-bar truss problem using the ε-constraint method

	w_1	w_2	f_1	f_2
1	0.00	1.00	15 930.33838658	7.19950556
2	0.05	0.95	20 914.68816803	5.13155444
3	0.10	0.90	25 864.58131198	4.03807309
4	0.15	0.85	30 826.29426326	3.35040831
5	0.20	0.80	35 780.28295944	2.86203497
6	0.25	0.75	40 749.20256135	2.49573187
7	0.30	0.70	45 711.63529606	2.24354812
8	0.35	0.65	50 669.82688126	2.01375450
9	0.40	0.60	55 605.63317530	1.80415992
10	0.45	0.55	60 572.65933753	1.67021785
11	0.50	0.50	65 531.69581605	1.57002894
12	0.55	0.45	70 403.35357416	1.55782006
13	0.60	0.40	75 437.07387688	1.49392476
14	0.65	0.35	80 385.88613412	1.45750160
15	0.70	0.30	85 354.06638307	1.41829067
16	0.75	0.25	90 335.42369639	1.38315281
17	0.80	0.20	95 268.62154318	1.35688732
18	0.85	0.15	100 230.48283892	1.33750258
19	0.90	0.10	105 209.97445419	1.32262364
20	0.95	0.05	110 142.10348170	1.31111391
21	1.00	0.00	115 114.75134273	1.30340205

to increase the stiffness is to increase the cross sectional area. A small increase in the cross sectional area of some truss members (like the cross-sectional area of bar 5 in Fig. 8) has little impact on the displacement, but it has a big impact on the weight. Even after normalization, the objective corresponding to the displacement is still less sensitive to the changes of some of the cross sectional areas. Numerical experiments show that reducing the upper bound of the cross sectional area can alleviate this effect.

All the changes made above just increase the accuracy of the results. They do not change the nature and magnitude of the objectives.

6 Conclusions

This paper details a methodology to deal with biobjective optimization problems based on approximating the Pareto set and presenting it to the decision-maker. With this approximation which is obtained using only three optimizations (one at each extremity and one intermediate), the decision-maker can observe the trade-off between the objectives and decide where on the Pareto curve the "optimal" solution lies. Then, using either the ε-constraint method or the goal programming approach, he/she can resolve the optimization to get accurate results. Thus, even in cases on nonconvex or disjointed Pareto sets, the approach proposed is a useful tool to the designer.

References

Athan, T.W.; Papalambros, P.Y. 1996: A note on weighted criteria methods for compromise solutions in multi-objective optimization. *Eng. Opt.* **27**, 155–176

Boychuk, L.M.; Ovchinnikov, V.O. 1973: Principal methods for solution of multicriteria optimization problems (survey). *Soviet Automatic Control* **6**, 1–4

Charnes, A.; Cooper, W.W. 1977: Goal programming and multiple objective optimization – Part I. *Euro. J. Oper. Res.* **1**, 39–54

Chen, W. 1995: *A robust concept exploration method for configuring complex systems*. Ph.D. Thesis, Georgia Institute of Technology

Das, I.; Dennis, J.E. 1997: A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multi-criteria problems. *Struct. Optim.* **14**, 63–69

Eschenauer, H.A. 1992: Multidisciplinary modeling and optimization in design processes. *ZAMM* **72**, 438–447

Fadel, G.M.; Li, Y.; Wiecek, M. 2002: Detecting on Local convexity on the Pareto surface. *J. Multi-Criteria Decision Analysis* (submitted)

Haftka, R.T.; Gurdal, Z. 1993: *Elements of structural optimization*, 3rd edn. Boston: Kluwer

Haimes, Y.Y.; Chankong, V. 1983: *Multiobjective decision making – theory and methodology*. New York: North Holland

Hwang, C.-L.; Masud, A.S.M. 1979: *Multiple objective decision making – methods and applications*. Berlin, Heidelberg, New York: Springer

Keeney, R.L.; Raiffa, H. 1999: *Decision with multiple objectives: preferences and value tradeoffs*. Cambridge University Press

Koski, J. 1985: Defectiveness of weighting method in multicriterion optimization of structures, *Comm. Appl. Numer. Meth.* **1**, 333–337

Li, Y. 1999: *Approximating the Pareto set of convex bi-criteria optimization problems to aid decision making in design*. Ph.D. Thesis, Clemson University

Li, Y.; Fadel, G.M. 1998: Approximating Pareto curves using the hyper-ellipse. *AIAA/ISSMO Internet Conf. on Multidisciplinary Optimization*

Lieberman, E.R. 1991: *Multi-objective programming in the USSR*. New York: Academic Press

Messac, A. 1996: Physical programming: effective optimization for computational design. *AIAA J.* **34**, 149–158

Messac, A.; Sundararaj, G.J.; Tapetta, R.V.; Renaud, J.E. 2000: The ability of objective functions to generate nonconvex Pareto frontiers. *AIAA J.* **38**, 1084–1191

Stadler, W. 1988: *Multicriteria optimization in engineering and in the sciences*. New York: Plenum Press

Stadler, W.; Dauer, J. 1993: Multicriteria optimization in engineering. In: Kamat, M.P. (ed.) *Structural optimization; status and promise*. Washington, D.C.: AIAA

Tind, J.; Wiecek, M.M. 1997: Augmented Lagrangian and Tchebycheff approaches in multiple objective programming. *J. Global Optimiz.* **14**, 251–266

Vanderplaats, Miura & Associates, Inc. 1993: *DOT user's manual, version 4.00*. Goleta, CA: VMA Engineering

Appendix

Results for the two-bar and ten-bar truss problems are given in Tables 13–16.