Engineering Optimization
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/geno20

GENERALIZED CENTER METHOD FOR MULTIOBJECTIVE ENGINEERING OPTIMIZATION

F. Y. CHENG [a] & X. S. LI [b]
[a] Department of Civil Engineering , University of Missouri-Rolla , Rolla, MO, 65409, USA
[b] Research Institute of Engineering Mechanics, Dalian University of Technology , Dalian, 116023, China
Published online: 27 Apr 2007.

Eng. Opt., 1999, Vol. 31, pp. 641–661
Reprints available directly from the publisher
Photocopying permitted by license only

GENERALIZED CENTER METHOD FOR MULTIOBJECTIVE ENGINEERING OPTIMIZATION

F. Y. CHENG[a,]* and X. S. LI[b]

[a] *Department of Civil Engineering, University of Missouri-Rolla, Rolla, MO 65409, USA;*
[b] *Research Institute of Engineering Mechanics, Dalian University of Technology, Dalian 116023, China*

(Received 14 October 1997; In final form 21 May 1998)

This paper presents a new approach to multiobjective engineering optimization: the generalized center method (GCM). A multiobjective problem is solved by calculating the centers of a sequence of level sets. These sets comprise intersections of the original constraints and level constraints imposed on objective functions. In view of the different dimensions and conflicting nature of multiple objectives, some scaling and trade-off procedures are implemented. Several engineering optimization examples are given to demonstrate the effectiveness of the proposed method.

Keywords: Multiobjective optimization; generalized center method; minimax problem; smooth approximation; Pareto optimal set; trade-off factor

1. INTRODUCTION

Multiobjective optimization techniques are a useful tool in engineering design to handle several conflicting objectives. Since the 1980s, a number of researchers in structural optimization area have pursued this technique [1–8]. Compared with traditional structural optimization, multiobjective optimization is still in the early stages of development. Further effort is therefore needed to refine problem formulations and solution techniques.

* Corresponding author.

A multiobjective optimization problem is usually posed as

$$\begin{aligned} &\min\ \mathbf{f}(\mathbf{x}) \equiv \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_l(\mathbf{x})\} \\ &s.t.\ g_j(\mathbf{x}) \leq 0 \quad j = 1, 2, \ldots, m \end{aligned} \tag{1}$$

where $\mathbf{x} \in R^n$ is a vector of decision variables; $f_i(\mathbf{x})$, $i = 1; 2, \ldots, l$, are objective functions; $g_j(\mathbf{x})$, $j = 1, 2, \ldots, m$, are constaint functions which determine a feasible region as

$$S \equiv \{\mathbf{x} \in R^n \| g_j(\mathbf{x}) \leq 0, \ \ j = 1, 2, \ldots, m\} \tag{2}$$

It should be noted that in multiobjective optimization, the definition of optimality is different from that for a single-objective case. To explore this difference, two basic concepts are pertinent.

Optimal Solution

$\mathbf{x}^* \in S$ is an optimal solution if and only if $f_i(\mathbf{x}^*) \leq f_i(\mathbf{x})$ for all $i = 1, 2, \ldots, l$.

In view of the conflicting nature of multiple objectives, an optimal solution generally does not exist. Thus it is necessary to be satisfied with compromise and non-inferior solutions, also known as Pareto optimal or efficient solutions.

Non-inferior (Pareto Optimal or Efficient) Solutions

$\mathbf{x}^* \in S$ is a non-inferior solution if and only if there does not exist any $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for some $i = 1, 2, \ldots, l$.

Normally, there is more than one non-inferior solution; a determination of the inferior set is not sufficient to make a decision. The systems analyst has to choose one solution which by some criterion is "best" or "preferred". Thus, additional criteria must be introduced to distinguish the "best" of the non-inferior solutions.

There are basically two approaches to the solution of multiobjective problems. First is to find the preferred solution directly. Second is to generate a set of non-inferior solutions and choose one among them. Frequently used techniques of multiobjective optimization include the weighting coefficient method, cooperative game theory method, ε-constraint method and goal programming method [9, 10]. For a multiobjective optimization problem, it is less difficult to find a non-inferior solution than to select the "best" compromise solution. It is well known that the search is easier than the selection. The generalized center method (GCM) proposed here is designed to fill the gap between single- and multiobjective optimization by integrating both search and selection in a natural way. It solves a multiobjective problem by calculating centers of a sequence of level sets. As noted, these sets comprise intersections of the original constraints and level constraints imposed on the objective functions, and therefore represent reduced feasible regions. To define the level constraints, moving targets are adopted instead of fixed goals or targets.

Advantages of GCM are as follows.

1. Inherits both theoretical and computational aspects of the well-established center method. It unifies single- and multiobjective optimization.
2. Obtains acceptable solutions by simultaneously minimizing all objectives. Some flexibility can be gained by adopting different trade-off factors so as to emphasize certain objectives.
3. Provides a compromise Pareto optimal solution with less computational effort. It is easy to implement and can be incorporated in the finite element based structural optimization package.

2. CENTER METHOD FOR SINGLE-OBJECTIVE OPTIMIZATION

GCM for multiobjective optimization is based on the center method for single objective optimization. The latter has been successfully used to solve linear and non-linear programming problems. It was first proposed by Huard [11] for solving non-linear constrained problems. With this method, a non-linear programming problem

$$\begin{aligned}&\min f(\mathbf{x})\\&s.t.\ g_j(\mathbf{x}) \le 0, \quad j = 1, 2, \ldots, m\end{aligned} \tag{3}$$

is transformed into an equivalent

$$\begin{aligned} &\min \alpha \\ &\quad s.t.\, f(\mathbf{x}) \le \alpha \\ &\qquad g_j(\mathbf{x}) \le 0, \quad j = 1, 2, \ldots, m \end{aligned} \tag{4}$$

where α represents an upper bound on the objective function. Instead of directly solving the problem by an algorithm, this method locates centers of a sequence of level sets. These level sets are defined as intersections of the original constraint set and a level constraint imposed on the objective function. If the original constraint set is denoted by X and the half space determined by the level constraint is denoted by

$$\Omega = \{\mathbf{x} \in R^n | f(\mathbf{x}) - \alpha \le 0\} \tag{5}$$

then the level set is defined by

$$\begin{aligned} S_\alpha = S \cap \Omega = \{\mathbf{x} \in R^n | f(\mathbf{x}) - \alpha \le 0;\ g_j(\mathbf{x}) \le 0, \\ j = 1, 2, \ldots, m\} \end{aligned} \tag{6}$$

where α is considered as a fixed parameter in each level set. It is apparent that $S_\alpha \subseteq S$, regardless of the value which parameter α has in a level set. In other words, level sets S_α are always contained in original constraint set S because any extra level constraint may eliminate part of the feasible region.

The basic theory of the center method is illustrated in Figure 1. When the iteration reaches an interior point $\mathbf{x}^\nu$ of S, the objective contour through $\mathbf{x}^\nu$ divides feasible region S into two parts. In one part, $f(\mathbf{x}) > f(\mathbf{x}^\nu)$ and in the other $f(\mathbf{x}) < f(\mathbf{x}^\nu)$. Obviously, much of feasible region S where $f(\mathbf{x}) > f(\mathbf{x}^\nu)$ (the shaded part of Fig. 1) need not be searched and is therefore removed from the next level set. This removal is accomplished by updating parameter α^ν. The simplest updating scheme is $\alpha^\nu = f(\mathbf{x}^\nu)$. For the next level set, the center should be in the unshaded area. Thus a decreasing sequence of parameter α^ν is established which makes level sets S_{α^ν} smaller and smaller, and subsequently the values of objective $f(\mathbf{x}^\nu)$ smaller and smaller. Eventually the centers of these level sets converge to the solution of the original problem.

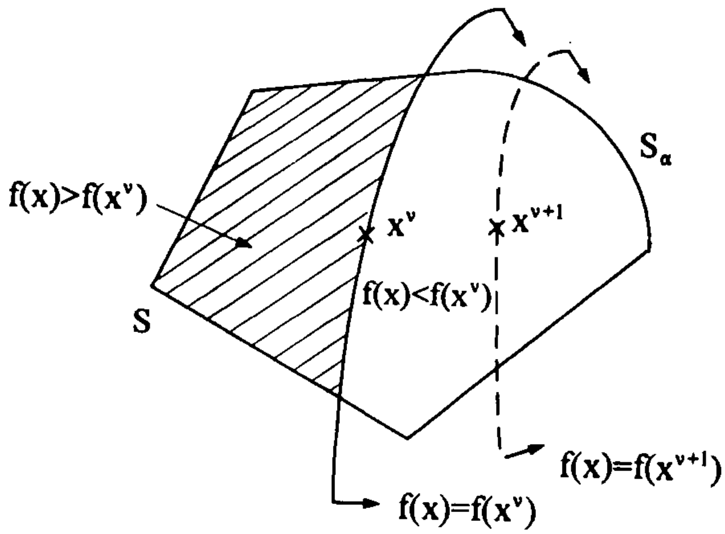

FIGURE 1 Center method for single-objective optimization.

For practical application, let $\alpha^{\nu} = rf(\mathbf{x}^{\nu})$, where $r > 1$ (or $r < 1$) if the objective function is to be minimized (or maximized). This allows the center $\mathbf{x}^{\nu}$ of current level set $S_{\alpha^{\nu}}$ to become the interior point of next level set $S_{\alpha^{\nu+1}}$. An important step in the center method is to calculate the center of level set S_{α} which is defined as the minimizer of some distance function of one of two forms, either

$$d_1(\mathbf{x}, \alpha) \equiv -\log[\alpha - f(\mathbf{x})] - \sum_{j=1}^{m} \log[-g_j(\mathbf{x})] \tag{7}$$

or

$$d_2(\mathbf{x}, \alpha) \equiv \max\{f(\mathbf{x}) - \alpha;\ \ g_j(\mathbf{x}),\ \ j = 1, \ldots, m\} \tag{8}$$

where $d_1(\mathbf{x}, \alpha)$ is a smooth function but needs a feasible initial point and requires that the iterations proceed within the feasible region while $d_2(\mathbf{x}, \alpha)$ has a simple form but discontinuous first derivatives.

In the generalized center method, $d_2(\mathbf{x}, \alpha)$ is adopted as the distance function. This means that it is necessary to solve a typical minimax problem. The non-smooth objective function $d_2(\mathbf{x}, \alpha)$ is replaced by an approximate function independently derived by Ben-Tal and Teboulle [12] and Li [13]. Thus the generalized center method solves the original problem by solving a sequence of unconstrained minimizations of the following function

$$d_p(\mathbf{x}, \alpha) = \frac{1}{p} \ln \left\{ \exp[p(f(\mathbf{x}) - \alpha)] + \sum_{j=1}^{m} \exp[p g_j(\mathbf{x})] \right\} \tag{9}$$

where $p > 0$ is a controlling parameter. It has been proved that $d_p(\mathbf{x}, \alpha) \rightarrow d_2(\mathbf{x}, \alpha)$ when $p \rightarrow \infty$. Details of the derivations and the properties of this approximate function are available in the literature cited earlier.

3. CENTER METHOD FOR MULTIOBJECTIVE OPTIMIZATION

It is relatively easy to apply the center method for single objective optimization to a multiobjective optimization problem such as that given by Eq. (1). If the half space defined by the level constraint on each objective is denoted by

$$\Omega_i \equiv \{\mathbf{x} \in R^n | f_i(\mathbf{x}) - \alpha_i \leq 0\}, \quad i = 1, \ldots, l \tag{10}$$

where α_i denotes the upper bound referred to as the "moving target" on objective $f_i(\mathbf{x})$, then the level set in the multiobjective case is defined by

$$\begin{aligned} S_\alpha \equiv \Omega_1 \cap \Omega_2 \cap \ldots \cap \Omega_l \cap S = \{\mathbf{x} \in R^n | f_i(\mathbf{x}) - \alpha_i \leq 0, \\ i = 1, \ldots, l; \;\; g_j(\mathbf{x}) \leq 0, \;\; j = 1, \ldots, m\} \end{aligned} \tag{11}$$

The center of a level set S_α can still be calculated by solving a minimax problem

$$\begin{aligned} \min_{\mathbf{x}} \;\; d(\mathbf{x}, \alpha) \equiv \max_{i,j} \{f_i(\mathbf{x}) - \alpha_i, \quad i = 1, \ldots, l; \;\; g_j(\mathbf{x}), \\ k = 1, \ldots, m\} \end{aligned} \tag{12}$$

Here level set S_α is the intersection of multiple half spaces $\Omega_i (i = 1, \ldots, l)$ and feasible region S. For simplification, assume that all the objectives are to be minimized. Level set S_α can now be reduced as in Figure 2. When the iteration reaches feasible point $\mathbf{x}^\nu$, each objective contour divides feasible region S into two parts. In one part $f_i(\mathbf{x}) > f_i(\mathbf{x}^\nu)$ and in the other $f_i(\mathbf{x}) < f_i(\mathbf{x}^\nu)$. To minimize the objectives, the undesirable part of S is eliminated and the part of S where $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^\nu)$ holds true for all the objectives is searched. As shown in Figure 2, the shaded area is excluded from subsequent iterations so that the level sets become progressively smaller.

Note that in multiobjective optimization, level set S_α can involve conflicting objectives. To avoid prematurely terminating the iteration and to put emphasis on certain objectives, trade-off factors can be applied to formulate moving target α_i. The simplest trade-off formula is

$$\alpha_i^\nu = r_i f_i(\mathbf{x}^\nu), \quad i = 1, 2, \ldots, l \tag{13}$$

where $r_i \geq 1$ denotes the trade-off factor for the ith objective function. This process is illustrated in Figure 3. Assume that only two objectives are to be minimized and that intersection $\Omega_1 \cap \Omega_2$ becomes empty, or

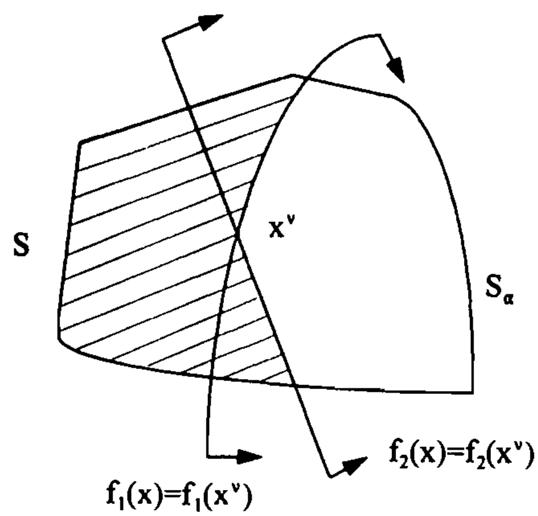

FIGURE 2 Center method for multiobjective optimization.

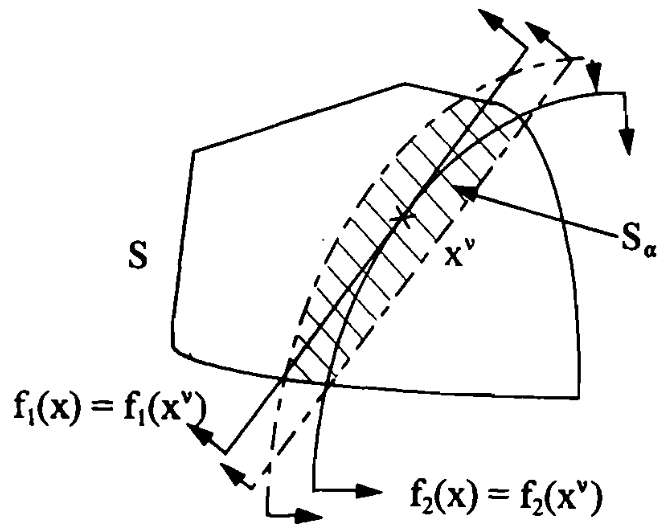

FIGURE 3 Trade-off in multiobjective optimization.

void, at point $\mathbf{x}^\nu$. Without trade-offs, computation would stop at $\mathbf{x}^\nu$ which might not yield a satisfactory compromise solution. With the trade-off given in Eq. (13), the two contours move apart and create a non-void level set, shown by the shaded area. Such an approach offers flexibility by assigning different trade-off factors to each objective.

As in the single-objective case, the non-smooth function $d(\mathbf{x}, \alpha)$ in Eq. (12) is replaced by its smooth approximation. The generalized center method then solves a multiobjective problem by solving a sequence of unconstrained minimizations as

$$\begin{aligned}\min_{\mathbf{x}} d_p(\mathbf{x}, \alpha^\nu) = \frac{1}{p} \ln \Bigg\{ & \sum_{l=1}^{l} \exp\left[p(f_l(\mathbf{x} - \alpha_l^\nu))\right] \\ & + \sum_{j=1}^{m} \exp[pg_j(\mathbf{x})] \Bigg\}\end{aligned} \tag{14}$$

where α_l^ν is updated by Eq. (13).

Since the objective functions in multiobjective optimization generally have different dimensions, they need to be scaled in some way. Here the following dimensionless objectives are used:

$$f_i^\nu(\mathbf{x}) = \frac{f_i(\mathbf{x}) - \alpha_i^\nu}{\| \alpha_i^\nu \|}; \quad i = 1, 2, \ldots, l \tag{15}$$

Correspondingly, level set S_α is now defined by

$$S_{\alpha^\nu} \equiv \{f_i^\nu(\mathbf{x}) \leq 0, \quad i = 1, \ldots, l; \ g_j(\mathbf{x}) \leq 0, \ j = 1, \ldots, m\} \tag{16}$$

and the distance function becomes

$$d(\mathbf{x}, \alpha^\nu) \equiv \max_{i,j} \{ f_i^\nu(\mathbf{x}), \quad i = 1, \ldots, l; \ g_j(\mathbf{x}), \ j = 1, \ldots, m\} \tag{17}$$

Furthermore, non-smooth function $d(\mathbf{x}, \alpha^\nu)$ is replaced by its smooth approximation

$$d_p(\mathbf{x}, \alpha^\nu) = \frac{1}{p} \ln \left\{ \sum_{i=1}^{l} \exp\left[p f_i^\nu(\mathbf{x})\right] + \sum_{j=1}^{m} \exp\left[p g_j(\mathbf{x})\right] \right\} \tag{18}$$

At each iteration GCM thus performs an unconstrained minimization of $d_p(\mathbf{x}, \alpha^\nu)$.

For a gradient-based unconstrained minimization algorithm, the first derivatives of $d_p(\mathbf{x}, \alpha^\nu)$ are written as

$$\frac{\partial d_p(\mathbf{x}, \alpha^\nu)}{\partial x_k} = \sum_{i=1}^{l} \lambda_i \frac{\partial f_i^\nu(\mathbf{x})}{\partial x_k} + \sum_{j=1}^{m} \lambda_j \frac{\partial g_j(\mathbf{x})}{\partial x_k}, \quad k = 1, \ldots, n \tag{19}$$

in which

$$\begin{aligned} \lambda_i &= \exp[p f_i^\nu(\mathbf{x})]/Z, \quad i = 1, \ldots, l \\ \lambda_j &= \exp[p g_j(\mathbf{x})]/Z, \quad j = 1, \ldots, m \end{aligned} \tag{20}$$

where

$$Z = \sum_{i=1}^{l} \exp\left[p f_i^\nu(\mathbf{x})\right] + \sum_{j=1}^{m} \exp\left[p g_j(\mathbf{x})\right] \tag{21}$$

Generalized Center Algorithm

Four steps comprise the process.

a. Initialize the iteration with $\mathbf{x}^0$ and choose a sufficiently large α^0; let $\nu = 0$.
b. Let $\nu = \nu + 1$; carry out the minimization of $d_p(\mathbf{x}, \alpha^\nu)$.
c. Check the feasibility of $\mathbf{x}^\nu$; if satisfied, then update $\alpha_i^\nu = r_i f_i(\mathbf{x}^\nu)$.
d. Check convergence, if satisfied, then stop; otherwise go to Step b. To save computational effort, the minimizations of $d_p(\mathbf{x}, \alpha^\nu)$ need not be carried out precisely.

4. ENGINEERING OPTIMIZATION EXAMPLES

To illustrate the generalized center method, several engineering design examples are presented. Computational results from this method are compared with those from other methods.

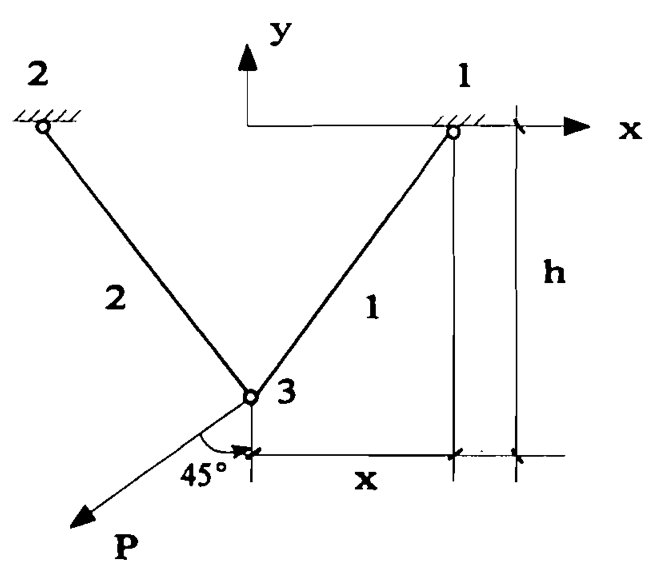

FIGURE 4 Two-bar truss.

Example 1 Two-Bar Truss

This example is taken from Rao's work [7]. He designed a two-bar truss shown in Figure 4 by means of game theory, with structural weight and resultant displacement of joint 3 as objective functions subject to stress constraints on the members and minimum limits on the design variables. Area A of members and position x of joints 1 and 2 serve as the design variables. The truss is symmetric. This problem is formulated as

$$
\begin{aligned}
\min\ & f_1(\mathbf{x}) = 2\rho h x_2\sqrt{1+x_1^2} \\
& f_2(\mathbf{x}) = \frac{Ph(1+x_1^2)^{1.5}(1+x_1^4)^{0.5}}{2\sqrt{2}Ex_1^2x_2} \\
s.t.\ & g_1(\mathbf{x}) = \frac{P(1+x_1)(1+x_1^2)^{0.5}}{2\sqrt{2}x_1x_2} - \sigma_0 \le 0 \\
& g_2(\mathbf{x}) = \frac{P(-x_1+1)(1+x_1^2)^{0.5}}{2\sqrt{2}x_1x_2} - \sigma_0 \le 0 \\
& g_3(\mathbf{x}) = x_1^{(l)} - x_1 \le 0 \\
& g_4(\mathbf{x}) = x_2^{(l)} - x_2 \le 0 \\
& g_5(\mathbf{x}) = x_1 - x_1^{(m)} \le 0 \\
& g_6(\mathbf{x}) = x_2 - x_2^{(m)} \le 0
\end{aligned}
\tag{22}
$$

where $x_1 = x/h$, $x_2 = A/A_{\min}$, E = Young's modulus, ρ = density of material, and $x_i^{(l)}$ = lower bound and $x_i^{(m)}$ = upper bound on design *variables* x_i, $i = 1, 2$. Problem data are $\rho = 0.283\,\text{lb/in}^3$, $h = 100\,\text{in}$, $P = 10^4\,\text{lb}$, $E = (3)10^7\,\text{lb/in}^2$, $\sigma_0 = (2)10^4\,\text{lb/in}^2$, $A_{\min} = 1\,\text{in}^2$, $x_1^{(l)} = 0.1$, $x_2^{(l)} = 0.5$, $x_1^{(m)} = 2.25$ and $x_2^{(m)} = 2.5$.

The generalized center method solves this problem by calculating centers of a sequence of level sets defined by

$$
\begin{aligned}
\bar{f}_1(\mathbf{x}) &= \frac{f_1(\mathbf{x})}{\alpha_1} - 1 = \frac{2\rho h x_2\sqrt{1+x_1^2}}{\alpha_1} - 1 \le 0 \\
\bar{f}_2(\mathbf{x}) &= \frac{f_2(\mathbf{x})}{\alpha_2} - 1 = \frac{Ph(1+x_1^2)^{1.5}(1+x_1^4)^{0.5}}{2\sqrt{2}Ex_1^2x_2\alpha_2} - 1 \le 0
\end{aligned}
$$

$$
\begin{aligned}
\bar{f}_3(\mathbf{x}) &= \frac{g_1(\mathbf{x})}{\sigma_0} = \frac{P(1+x_1)(1+x_1^2)^{0.5}}{2\sqrt{2}x_1x_2\sigma_0} - 1 \leq 0 \\
\bar{f}_4(\mathbf{x}) &= \frac{g_2(\mathbf{x})}{\sigma_0} = \frac{P(-x_1+1)(1+x_1^2)^{0.5}}{2\sqrt{2}x_1x_2\sigma_0} - 1 \leq 0 \\
\bar{f}_5(\mathbf{x}) &= \frac{g_3(\mathbf{x})}{x_1^{(l)}} = 1 - \frac{x_1}{x_1^{(l)}} \leq 0 \\
\bar{f}_6(\mathbf{x}) &= \frac{g_4(\mathbf{x})}{x_2^{(l)}} = 1 - \frac{x^2}{x_2^{(l)}} \leq 0 \\
\bar{f}_7(\mathbf{x}) &= \frac{g_5(\mathbf{x})}{x_1^{(m)}} = \frac{x_1}{x_1^{(m)}} - 1 \leq 0 \\
\bar{f}_8(\mathbf{x}) &= \frac{g_6(\mathbf{x})}{x_2^{(m)}} = \frac{x_2}{x_2^{(m)}} - 1 \leq 0
\end{aligned} \tag{23}
$$

where α_1, α_2 are moving targets for objective functions $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$, respectively, and are to be updated at each iteration.

To find the center of the level set defined by Eq. (23), the generalized center method sets up the following minimax problem

$$\min_{\mathbf{x}} \bar{f}(\mathbf{x}) \equiv \max_{1 \leq i \leq 8} \{\bar{f}_i(\mathbf{x})\} \tag{24}$$

where $\bar{f}(\mathbf{x})$ is the distance function. In place of non-smooth $\bar{f}(\mathbf{x})$, the following approximate function is substituted

$$\bar{f}_p(\mathbf{x}) = \frac{1}{p} \ln \sum_{i=1}^{8} \exp[p\bar{f}_i(\mathbf{x})] \tag{25}$$

where p is a controlling parameter and is usually designated as $10^3 \sim 10^4$.

The iteration starts from $\{x_1, x_2\} = \{5, 5\}$ with trade-off factors for both objectives designated as 1.0. Convergence is achieved after two iterations. Table I lists the iteration history of these calculations. Table II compares results of GCM with those obtained by the game theory approach [7]. The second and third columns list the results by minimizing $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ separately. Comparing the fourth and fifth columns shows how close the results are.

TABLE I Iteration history for two-bar truss

Iteration No. (v)	0	1	2
f_1(Weight)	1443.0225	89.5064	83.9268
f_2(Displacement)	0.3127	0.0422	0.0395
$x_1 = x/h$	5.0000	1.0000	0.7680
$x_2 = A/A_{\min}$	5.0000	1.1180	1.1760
Stress for Bar 1, σ_1	4326.662	8944.272	8726.426
Stress for Bar 2, σ_2	−2884.442	0.010	1145.350

TABLE II Comparison of results from two methods

	Minimization of f_1	*Minimization of* f_2	*Game Theory*	*GCM*
f_1	36.1473	186.7361	81.4137	83.9268
f_2	0.0943	0.0182	0.0408	0.0395
$x_1 = x/h$	0.6473	0.8612	0.7681	0.7680
$x_2 = A/A_{\min}$	0.5295	2.5000	1.1408	1.1760
σ_1	19994.9	4033.6	8996.0	8726.4
σ_2	3889.7	300.9	1180.1	1145.4

Example 2 Four-Bar Truss

This example was studied by Stadler [8] who gave a detailed analysis of the four-bar truss shown in Figure 5. It is designed with structural volume and joint displacement Δ as objective functions subject to four constraints related to member stresses. Areas of member cross-sections serve as the design variables. Mathematically, this problem is posed as

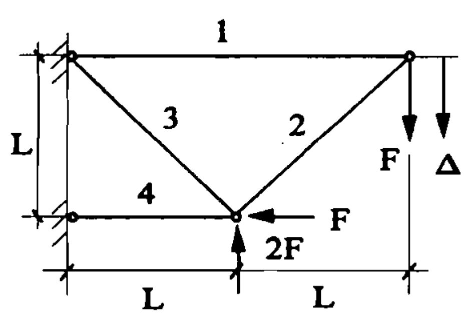

FIGURE 5 Four-bar truss.

$$
\begin{aligned}
\min f_1(\mathbf{x}) &= L\left(2x_1 + \sqrt{2}x_2 + \sqrt{x_3} + x_4\right) \\
f_2(\mathbf{x}) &= \frac{FL}{E}\left(\frac{2}{x_1} + \frac{2\sqrt{2}}{x_2} - \frac{2\sqrt{2}}{x_3} + \frac{2}{x_4}\right) \\
s.t.\quad & (F/\sigma) \le x_1 \le 3(F/\sigma) \\
& \sqrt{2}(F/\sigma) \le x_2 \le 3(F/\sigma) \\
& \sqrt{2}(F/\sigma) \le x_3 \le 3(F/\sigma) \\
& (F/\sigma) \le x_4 \le 3(F/\sigma)
\end{aligned}
\tag{26}
$$

where $F = 10\,\text{kN}$, $E = (2)10^5\,\text{kN/cm}^2$, $L = 200\,\text{cm}$, $\sigma = 10\,\text{kN/cm}^2$.

Stadler showed that $x_3 = \sqrt{2}(F/\sigma)$ was a constant, and the entire Pareto optimal set was composed of three line segments in 3-D space $(x_1 - x_2 - x_4)$ as depicted in Figure 6. These segments can be expressed by the following parametric equations.

Segment $1-2$

$$
\begin{gathered}
x_1 = F/\sigma,\ x_2 = \sqrt{2}F/\sigma,\ x_3 = \sqrt{2}F/\sigma, \\
x_4 = \sqrt{2}t,\ \text{with } t \ge \frac{\sqrt{2}}{2}F/\sigma
\end{gathered}
\tag{27}
$$

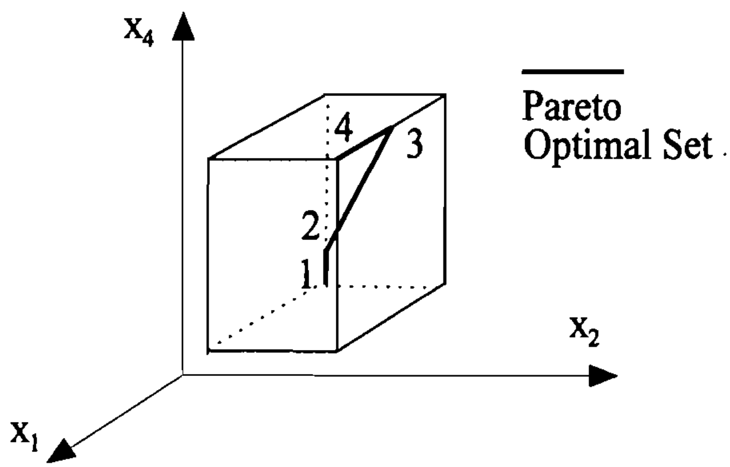

FIGURE 6 Pareto optimal set for four-bar truss.

Segment 2 – 3

$$x_1 = t,\ x_2 = \sqrt{2}t,\ x_3 = \sqrt{2}F/\sigma,$$
$$x_4 = \sqrt{2}t, \text{ with } F/\sigma \leq t \leq \frac{3\sqrt{2}}{2}F/\sigma \tag{28}$$

Segment 3 – 4

$$x_1 = t,\ x_2 = 3F/\sigma,\ x_3 = \sqrt{2}F/\sigma,\ x_4 = 3F/\sigma,$$
$$\text{with } \frac{3\sqrt{2}}{2}F/\sigma \leq t \leq 3F/\sigma \tag{29}$$

Table III lists computational results by the generalized center method. The iteration starts from $\mathbf{x} = [5, 5, 5, 5]^T$ and converges after seven iterations, attaining a solution at $\bar{\mathbf{x}} = [1.4128, 1.9980, 1.4149, 1.9980]^T$ with $\bar{f}_1 = 1930.0380$ and with $\bar{f}_2 = 0.0183$. Figure 7 draws a Pareto curve in the image ($f_1 \sim f_2$) space based on Eqs. (26 – 29) and shows how the iterations reach a Pareto optimal solution. Iteration points are drawn from the data computed in Table III. Note that the iterations move directly to a Pareto solution, and the convergence is quite smooth.

Example 3 Twenty-Five Bar Truss

This example, shown in Figure 8, is designed with structural weight W and virtual energy E_w as objectives to be minimized, subject to stress and size constraints. The truss is assumed to be symmetric and all the members are divided into eight groups: A_1, $A_2 = A_3 = A_4 = A_5$,

TABLE III Iteration history of four-bar truss

Iter. No. (ν)	f_1	f_2	x_1	x_2	x_3	x_4
0	5828.4271	0.0080	5	5	5	5
1	2219.5793	0.0212	1.8863	1.9230	1.9230	1.8863
2	2006.6614	0.0192	1.4480	2.0479	1.5509	2.0478
3	1951.5133	0.0186	1.4225	2.0117	1.4538	2.0117
4	1935.5753	0.0184	1.4149	2.0010	1.4265	2.0010
5	1931.0525	0.0184	1.4129	1.9981	1.4182	1.9981
6	1930.0079	0.0183	1.4126	1.9977	1.4157	1.9977
7	1930.0380	0.0183	1.4128	1.9980	1.4149	1.9980

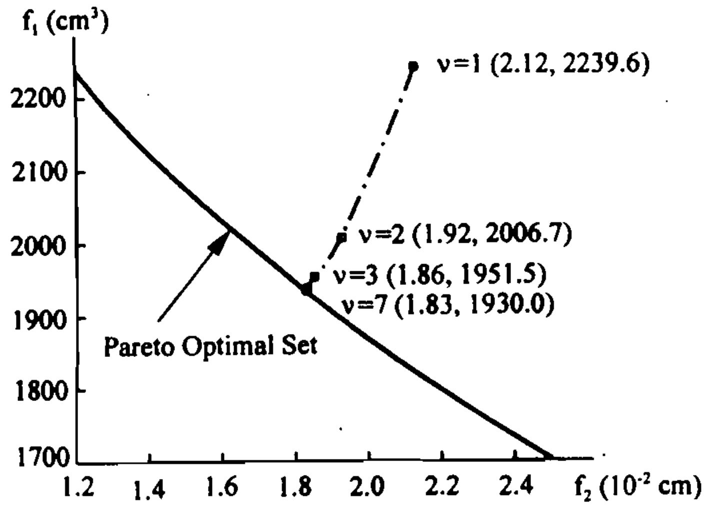

FIGURE 7 Convergence process in objective space.

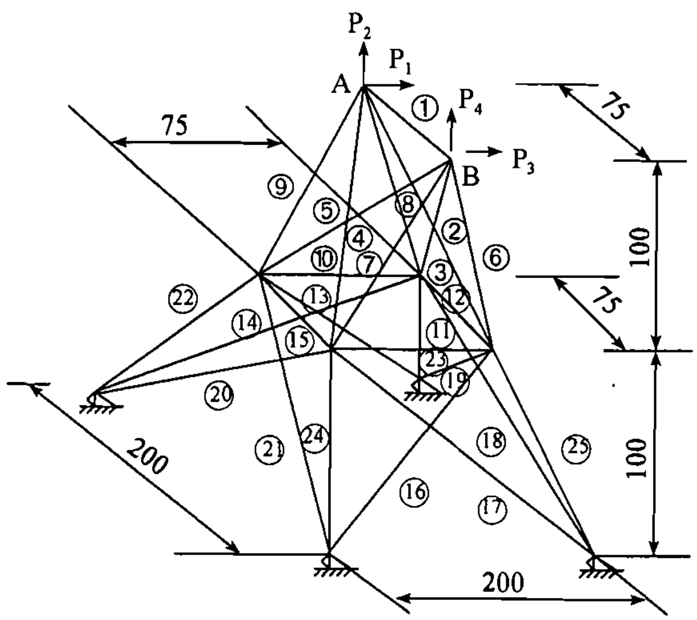

FIGURE 8 Twenty-five-bar truss.

$A_6 = A_7 = A_8 = A_9, A_{10} = A_{11}, A_{12} = A_{13}, A_{14} = A_{15} = A_{16} = A_{17}, A_{18}, =$ $A_{19} = A_{20} = A_{21}$ and $A_{22} = A_{23} = A_{24} = A_{25}$. One loading condition is considered, and applied loads are $P_{1y} = 20$ kips, $P_{1z} = -5$ kips, $P_{2y} = -20$ kips and $P_{2z} = -5$ kips, where the first subscript denotes the joint number and the second denotes the acting direction of loads.

Mathematically, this problem is formulated as

$$
\begin{aligned}
\min \ W &= \sum_{i=1}^{n} \rho_1 l_1 A_1 \\
E_w &= \sum_{i=1}^{n} \frac{\sigma \bar{\sigma}_i v_i}{E_i} \\
s.t. \ \sigma_i &\leq \sigma_i^0, \qquad i = 1, 2, \ldots, n \\
A_i &\geq A_{\min}, \quad i = 1, 2, \ldots, n
\end{aligned} \tag{30}
$$

where A_i is cross-section area, l_i is length, v_i is volume, E_i is Young's modulus and ρ_i is density of the ith member; σ_i and $\bar{\sigma}_i$ are real and virtual stresses of the ith member, respectively. The dummy load $\bar{P} = 1$ is applied at y-direction of joint 1. Other design parameters are $E_i = 10^4$ ksi, $\rho_i = 0.01$ lb/in^3, $\sigma_i^0 = 25$ ksi and $A_{\min} = 0.1$ in.

The iteration starts from infeasible point $A = [1, 1, 1, 1, 1, 1, 1, 1]^T$, and reaches point $W = 28.723$ lb and $E_w = 0.497$ in by five iterations, compared to the solution $W = 40.611$ lb and $E_w = 0.339$ in obtained by Li [14]. Note that the solution from the generalized center method is close to a Pareto optimal solution, as shown in Figure 9. The curve drawn for the Pareto optimal set is based on single-objective optimizations in which weight W serves as the objective function and virtual energy E_w as an additional constraint with different upper limits. The iteration history is shown in Table IV.

Example 4 Water Resource Planning
The problem involves optimal planning for a storm drainage system in an urban area, described by Musselman and Talavage [15]. Three decision variables are assumed for the storm drainage system: x_1 = local detention storage capacity, x_2 = maximum treatment rate, and x_3 = maximum allowable overflow rate. Objective functions to be minimized are f_1 = drainage network cost, f_2 = storage facility cost,

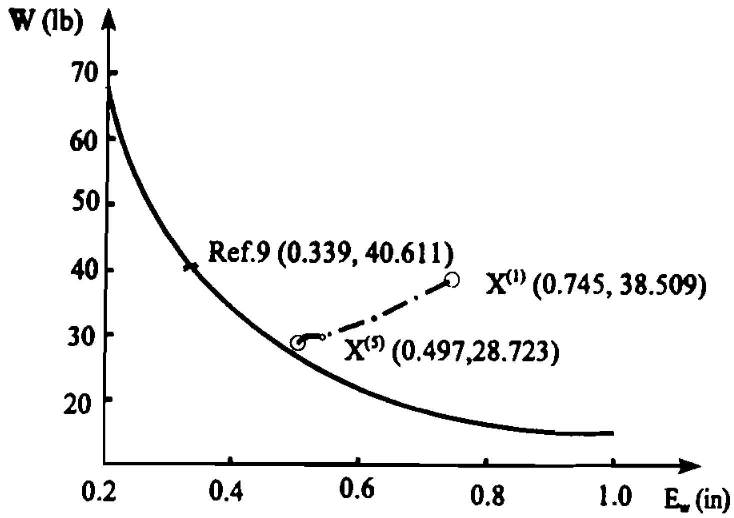

FIGURE 9 Pareto optimal solution for twenty-five-bar truss.

TABLE IV Computational results for twenty-five bar truss

Iter. No.	1	2	3	4	5
W	38.509	32.706	29.650	29.169	28.723
E_w	0.745	0.607	0.538	0.506	0.497
A_1	1.030	0.601	0.277	0.100	0.100
A_2	1.071	1.403	1.669	1.701	1.581
A_3	1.236	1.318	1.605	1.777	1.850
A_4	1.410	0.804	0.343	0.101	0.100
A_5	1.410	0.804	0.343	0.101	0.100
A_6	0.186	0.100	0.100	0.152	0.216
A_7	0.782	1.057	1.179	1.312	1.310
A_8	2.927	1.593	0.671	0.321	0.212

f_3 = treatment facility cost, f_4 = expected flood damage cost, and f_5 = expected economic loss due to flood. The mutiobjective formulation of this problem is given as

$$\min_{x\in\Omega} \; f_1 = 106780.37(x_2 + x_3) + 61704.67$$

$$f_2 = 3000x_1$$

$$f_3 = \frac{(305700)2289x_2}{[(0.06)2289]^{0.65}}$$

$$
\begin{aligned}
f_4 &= (250)2289\exp(-39.75x_2 + 9.9x_3 + 2.74) \\
f_5 &= 25\left(\frac{1.39}{x_1x_2} + 4940x_3 - 80\right) \\
s.t.\ g_1 &= \frac{0.00139}{x_1x_2} + 4.94x_3 - 0.08 \leq 1 \\
g_2 &= \frac{0.000306}{x_1x_2} + 1.082x_3 - 0.0986 \leq 1 \\
g_3 &= \frac{12.307}{x_1x_2} + 49408.24x_3 + 4051.02 \leq 50000 \\
g_4 &= \frac{2.098}{x_1x_2} + 8046.33x_3 - 696.71 \leq 16000 \\
g_5 &= \frac{2.138}{x_1x_2} + 7883.39x_3 - 705.04 \leq 10000 \\
g_6 &= \frac{0.417}{x_1x_2} + 1721.26x_3 - 136.54 \leq 2000 \\
g_7 &= \frac{0.164}{x_1x_2} + 631.13x_3 - 54.48 \leq 550
\end{aligned}
\tag{31}
$$

where $\Omega = \{0.01 \leq x_1 \leq 0.45;\ 0.01 \leq x_2 \leq 0.10;\ 0.01 \leq x_3 \leq 0.10\}$.

The iteration starts from $\mathbf{x} = [0.45, 0.1, 0.1]^T$ and reaches a compromise solution, shown in the last column of Table V, by three iterations. As verified by the ε-constraint method, the solution is a Pareto optimal solution. For verification, one of the five objectives was chosen as the objective function and the other four as additional constraints with upper limits fixed at the values found by the generalized center method. Then a single-objective optimization was performed obtaining an objective value nearly identical to that calculated by the GCM technique.

TABLE V Computational results for water resource planning

Iter. No.	0	1	2	3
f_1	0.830607(10^5)	0.669022(10^5)	0.660927(10^5)	0.660739(10^5)
f_2	0.135000(10^4)	0.425640(10^3)	0.409851(10^3)	0.408571(10^3)
f_3	0.285347(10^7)	0.899667(10^6)	0.874761(10^6)	0.877290(10^6)
f_4	0.447904(10^6)	0.299905(10^7)	0.290543(10^7)	0.288762(10^7)
f_5	0.111222(10^5)	0.788586(10^4)	0.758636(10^4)	0.755560(10^4)
x_1	0.45	0.1419	0.1366	0.1362
x_2	0.10	0.0315	0.0307	0.0307
x_3	0.10	0.0171	0.0104	0.0102

5. CONCLUSIONS

Multiobjective optimization problems can usually be solved by the generalized center method in the same manner as single-objective problems. This method thus unifies single and multiple objective optimization. From a computational viewpoint, the method has the ability to effectively find a compromise Pareto optimal solution with less programming and computational effort.

For illustrative purposes, computational results for the four examples were obtained with all trade-off factors uniform with a unit value. In practice, one may emphasize certain objectives by assigning different values of trade-off factors. By doing so, different compromise solutions can be expected. While the simple trade-off scheme presented here works well, a more sophisticated trade-off scheme may be needed in some applications.

Overall, the generalized center method offers a promising alternative for multiobjective optimization. Further refinement of the method would be a worthwhile research task.

Acknowledgments

This research is based on a project partially supported by the US National Science Foundation under grant NSF CMS 9703725 at the University of Missouri-Rolla and by State-Key Laboratory of Structural Analysis for Industrial Equipment in China. The support is gratefully acknowledged.

References

[1] Borri, A. and Speranizi, E. (1995). The use of an F.E. code for multi-criteria reliability-based optimization of composite material structure, *Reliability and Optimization of Structural Systems*. Rackwitz, R., Augusti, G. and Borri, A. (eds.), Chapman & Hill, London, pp. 88–95.

[2] Cheng, F. Y. and Li, D. (1996). Multiobjective optimization of structures with and without control. *Journal of Guidance, Control and Dynamics*, 19(2), 392–397.

[3] Duckstein, L. (1984). Multiobjective optimization in structural design: the model choice problem. *New Directions in Optimum Structural Design*, Atrek, E., Gallagher, R. H., Ragsdell, K. M. and Zienkiewicz, O. C. (eds.), John Wiley & Sons, New York, pp. 459–481.

[4] Eschenauer, H., Koski, J. and Osyczka, A. (1990). *Multicriteria Design Optimization – Procedures and Applications*. Springler-Verlag, Berlin.

[5] Grandhi, R. V., Bharatram, G. and Venkayya, V. B. (1993). Multiobjective optimization of large-scale structures. *AIAA Journal*, **31**(7), 1329–1337.
[6] Koski, J. (1984). Multicriterion optimization in structural design. *New Directions in Structural Design*, Atrek, E., Gallagher, R. H., Ragsdell, K. M. and Zienkiewicz, O. C. (eds.), John Wiley & Sons, New York, pp. 485–503.
[7] Rao, S. S. (1987). Game theory approach for multiobjective structural optimization. *Computers and Structures*, **25**(1), 119–127.
[8] Stadler, W. and Dauer, J. (1992). Multicriteria optimization in engineering: A tutorial and survey. *Structural Optimization: Status and Future*, American Institute of Aeronautics and Astronautics, pp. 209–249.
[9] Osyczka, A. (1984). *Multicriterion Optimization in Engineering*. Ellis Horwood, Chichester.
[10] Ringuest, J. L. (1992). *Multiobjective Optimization – Behavioral and Computational Considerations*. Kluwer Academic, Boston.
[11] Huard, P. (1967). Resolution of mathematical programming with nonlinear constraints by the method of centers. *Nonlinear Programming*, Abadie, J. (ed.), North-Holland, Amsterdam, pp. 207–219.
[12] Ben-Tal, A. and Teboulle, M. (1989). A smooth technique for nondifferentiable optimization problems. *Lecture Notes in Mathematics*, **1405**, 1–11.
[13] Li, X. S. (1992). An entropy-based aggregate method for minimax optimization. *Engineering Optimization*, **18**, 277–285.
[14] Li, Dan (1997). Multiobjective optimization design of static and seismic-resistant structures with genetic algorithm, fuzzy logic and game theory. *Ph.D. Dissertation*, Department of Civil Engineering, University of Missouri-Rolla, USA.
[15] Musselman, K. and Talavage, J. (1980). A trade-off cut approach to multiple objective optimization. *Operations Research*, **28**(6), 1424–1435.