JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 74, No. 1, JULY 1992

Reference Point Approximation Method for the Solution of Bicriterial Nonlinear Optimization Problems

J. JAHN[1] AND A. MERKEL[2]

Communicated by W. Stadler

Abstract. This paper presents a reference point approximation algorithm which can be used for the interactive solution of bicriterial nonlinear optimization problems with inequality and equality constraints. The advantage of this method is that the decision maker may choose arbitrary reference points in the criteria space. Moreover, a special tunneling technique is given for the computation of global solutions of certain subproblems. Finally, the proposed method is applied to a mathematical example and a problem in mechanical engineering.

Key Words. Multi-objective optimization, interactive methods.

1. Introduction

Problems of bicriterial nonlinear optimization arise in different areas in the applied sciences, for instance structural mechanics, chemical processes, and chip design. Sometimes problems with more than two objectives can be transformed or even simplified to bicriterial problems. We assume that the constraints are given explicitly in the form of inequalities and equalities.

To be more specific, we have the following assumption.

Assumption 1.1. $\hat{S} \in \mathbb{R}^n$ is a given nonempty set; $f=(f_1, f_2): \hat{S} \to \mathbb{R}^2$, $g_i: \hat{S} \to \mathbb{R}$, $i=1,2,\ldots,p$, and $h_i: \hat{S} \to \mathbb{R}$, $i=1,2,\ldots,q$, are given functions.

[1]Associate Professor, Institut für Angewandte Mathematik, Universität Erlangen-Nürnberg, Erlangen, Germany.

[2]Research Assistant, Institut für Angewandte Mathematik, Universität Erlangen-Nürnberg, Erlangen, Germany.

0022-3239/92/0700-0087$06.50/0

Under this assumption, we consider the bicriterial optimization problem

$$\min \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix}, \tag{1a}$$

$$\text{s.t.} \quad g_i(x) \leq 0, \; i = 1, 2, \ldots, p, \tag{1b}$$

$$h_i(x) = 0, \; i = 1, 2, \ldots, q, \tag{1c}$$

$$x \in \hat{S}. \tag{1d}$$

Notice that we have to minimize the criteria f_1 and f_2 in the sense of Edgeworth–Pareto (a definition of a minimal solution is given below). If the resulting functions are defined on the whole space $\mathbb{R}^n$, then very often the set $\hat{S}$ can be chosen as $\mathbb{R}^n$. For simplicity, we define the constraint set of the problem (1) as

$$S := \{x \in \hat{S} \mid g_i(x) \leq 0, \; i = 1, 2, \ldots, p; \; h_i(x) = 0, \; i = 1, 2, \ldots, q\}.$$

Definition 1.1. Let Assumption 1.1 be satisfied. A vector $\bar{x} \in S$ is called a minimal solution, or an Edgeworth–Pareto optimal point, or a functional efficient point of the bicriterial optimization problem (1), if there does not exist any $x \in S$ with

$$f_1(x) \leq f_1(\bar{x}),$$

$$f_2(x) \leq f_2(\bar{x}),$$

where the strict inequality sign holds for at least one inequality. The set of minimal solutions of the problem (1) is denoted by M.

Of course we assume that the problem (1) is solvable, or in other words that the assumption below holds.

Assumption 1.2. The set M of minimal solutions of the problem (1) is nonempty.

The mathematical vector optimization problem (1) would be solved, if we could determine the whole set M of minimal solutions; but the original decision problem which leads to the mathematical formulation (1) is not completely solved if the set M is determined. The decision maker wants to select a minimal solution among all elements of M. Such a subjectively optimal solution can be determined by a so-called interactive method which uses additional information about the preference structure of the decision maker.

Until today, there are various methods for the interactive solution of vector optimization problems (e.g., see Ref. 1 for classical approaches). In this paper, we use a reference point approximation method, which has already been successfully applied to linear vector optimization problems (see Ref. 2). The conception of this algorithm reads as follows.

Algorithm 1.1.

Step 0. The decision maker chooses a weighted Chebyshev norm $\|\cdot\|$.

Step 1. The decision maker chooses an arbitrary reference point $y^{(1)} \in \mathbb{R}^2$. Set $i=1$.

Step 2. Compute a solution of the approximation problem

$$\min_{x \in M} \|y^{(i)} - f(x)\|. \tag{2}$$

Step 3. This solution is presented to the decision maker, who terminates the iteration process or proceeds as follows: The decision maker chooses another reference point $y^{(i+1)}$ and continues the procedure with $i := i+1$ in Step 2.

If the decision maker applies this method to a vector optimization problem, he has to articulate his goals by choosing an arbitrary reference point $y^{(i)} \in \mathbb{R}^2$. Then, a minimal solution of the problem (1) is determined whose image is as close as possible to the goal vector of the decision maker in the sense of a weighted Chebyshev norm. One can choose also other norms, like the Euclidean norm, but the Chebyshev norm leads to a simpler interpretation of the results.

It should be mentioned that Algorithm 1.1 is related to other reference point methods (e.g., see Refs. 3 and 4). The main difference from the known reference point methods lies in the optimization problem in Step 2 of Algorithm 1.1: the constraint set is M and not S. So, a solution of this problem is always a minimal solution of the original problem (1), even if we choose the reference point $y^{(i)}$ arbitrarily. But notice also that the approximation problem (2) may not be solvable, because the set M does not need to be closed; in this case, it is better to replace "min" by "inf" in (2).

The following theorem shows that the approximation problem (2) can be simplified in special cases.

Theorem 1.1. See, e.g., Ref. 5, Corollary 3.1 (a). Let Assumption 1.1 be satisfied; let the vector optimization problem (1) be given; and assume that there is a reference point $y \in \mathbb{R}^2$ with

$$y_i < f_i(x), \qquad \text{for all } x \in S \text{ and } i = 1, 2.$$

Then, a vector $\bar{x} \in S$ is a minimal solution of the problem (1) if and only if there exist $t_1, t_2 > 0$ such that

$$\max_{i=1,2}\{t_i(f_i(\bar{x}) - y_i)\}$$
$$< \max_{i=1,2}\{t_i(f_i(x) - y_i)\}, \qquad \text{for all } x \in S \text{ with } f(x) \neq f(\bar{x}). \tag{3}$$

Inequality (3) says that $\bar{x}$ is an image-unique solution of the approximation problem

$$\min_{x \in S} \|y - f(x)\|,$$

where $\|\cdot\|$ denotes the weighted Chebyshev norm with weights t_1 and t_2. So, in the special case that the reference point has certain properties, the approximation problem (2) in Algorithm 1.1 can be simplified. But one should also see that a best approximation has to be image-unique, which is hard to check on a computer (usual efficiency tests lead to numerical difficulties).

2. Algorithm

Based on the conception of the reference point approximation method, we want to formulate a method which can be used in practice. The main difficulty which arises in Algorithm 1.1 is the determination of the set M of minimal solutions of the problem (1). This set can be approximately determined with the aid of a method due to Polak (Ref. 6), which is reviewed in a simplified form.

2.1. Modified Polak Method. For an appropriate approximation of the set M [and the image set $f(M)$, respectively], one determines the points

$$a := \min_{x \in S} f_1(x),$$

$$b := f_1(\bar{x}), \qquad \text{with } f_2(\bar{x}) = \min_{x \in S} f_2(x),$$

on the y_1-axis (see Fig. 1). Then, one discretizes the interval $[a, b]$ by choosing points

$$y_1^{(k)} := a + k(b-a)/m, \qquad \text{for } k = 0, 1, 2, \ldots, m, \text{ with } m \in \mathbb{N} \text{ fixed.}$$

For every discretization point $y_1^{(k)}$, $k = 0, 1, 2, \ldots, m$, one solves the scalar

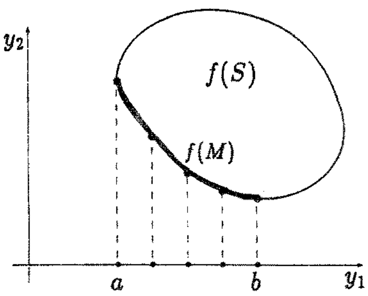

Fig. 1. Approximation of $f(M)$ with the Polak method.

optimization problem

$$\min \quad f_2(x), \tag{4a}$$

$$\text{s.t.} \quad x \in S, \tag{4b}$$

$$f_1(x) = y_1^{(k)}. \tag{4c}$$

Then, one gets points $y^{(k)} := f(x^{(k)})$ [where $x^{(k)}$ is a solution of the optimization problem (4)], which belong to the set $f(S)$. But in general, it is not ensured that the preimages $x^{(k)}$ are in fact minimal solutions of the vector optimization problem (2).

In order to obtain minimal solutions, one has to select points such that

$$y_2^{(k_1)} > y_2^{(k_2)} > y_2^{(k_3)} > \cdots$$

describes a strictly monotonically decreasing sequence. So, the set $\{x^{(k_1)}, x^{(k_2)}, \ldots, x^{(k_{m'})}\}$ is an approximation of the set M of minimal solutions (see Fig. 2). In this case, the approximation problem

$$\min_{x \in M} \|y - f(x)\|$$

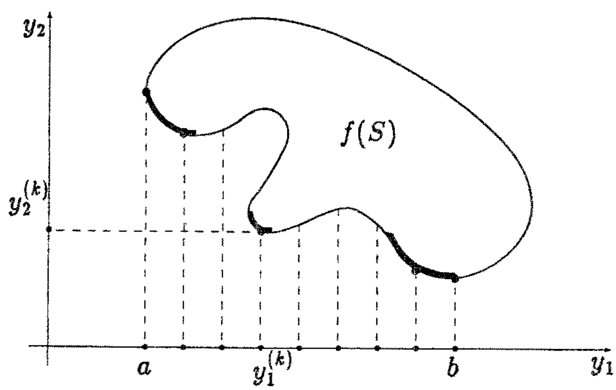

Fig. 2. Determination of minimal elements of $f(S)$.

in Step 2 of Algorithm 1.1 (where $y \in \mathbb{R}^2$ is now an arbitrary reference point) can be replaced by the simpler problem

$$\min_{j=1,2,\ldots,m'} \|y - f(x^{(k_j)})\|.$$

If the set $f(M)$ is connected, Polak uses splines in order to connect the computed points $f(x^{(k_j)})$. Here, we connect these points by straight lines, because the boundary part $f(M)$ of the set $f(S)$ does not need to be smooth.

For the solution of the scalar optimization problem (4), one needs a constrained optimization method, for instance a usual penalty method. Since most of these methods can be used for the calculation of only local minima and not for the determination of global minima, in general the Polak method cannot be carried out satisfactorily. So, for the solution of practical problems, one has to guarantee that the determined "solutions" of problem (4) are in fact also global solutions.

2.2. Tunneling Technique. In this section, we turn our attention to the determination of the global solutions of the subproblem (4) for $\hat{S} = \mathbb{R}^n$ in Assumption 1.1. If f_1 and f_2 are nonlinear functions, we have to apply a method for nonlinear constrained optimization. The usual algorithms of this class can be used only for the determination of local solutions. Methods designed for the computation of global solutions (for instance a Monte Carlo method) are time consuming. Since the parametric optimization problem (4) has to be solved for every parameter $y_1^{(0)}, y_1^{(1)}, \ldots, y_1^{(m)}$, one needs a numerical method which is not too slow and which guarantees that the obtained solutions are global ones. Therefore, we use a hybrid method, which is a combination of a penalty method and a tunneling technique.

If one uses a penalty method for the solution of the constrained optimization problem (4), with $\hat{S} = \mathbb{R}^n$, one has to solve a sequence of unconstrained problems of the form

$$\min_{x \in \mathbb{R}^n} f_2(x) + \mu_j p(x), \tag{5}$$

where $(\mu_j)_{j \in \mathbb{N}}$ is a sequence of positive real numbers tending to infinity and p is a penalty function i.e., especially,

$$p(x) \geq 0, \text{ for all } x \in \mathbb{R}^n, \text{ and } p(x) = 0 \Leftrightarrow x \in S \wedge f_1(x) = y_1^{(k)};$$

e.g., see Ref. 7. Assume that $\hat{x} \in \mathbb{R}^n$ is an approximation of a solution of the problem (5) (in practice, it is obtained by applying a computer program). Then, for some fixed $\epsilon > 0$, consider the constrained optimization problem

$$\min \quad 1/[f_2(\hat{x}) - f_2(x)] + \mu_j p(x), \tag{6a}$$

$$\text{s.t.} \quad f_2(\hat{x}) - f_2(x) \geq \epsilon, \tag{6b}$$

$$x \in \mathbb{R}^n. \tag{6c}$$

A solution $\bar{x}$ of this problem has the property that

$$f_2(\bar{x}) \leq f_2(\hat{x}) - \epsilon < f_2(\hat{x}),$$

and by the minimization of $1/[f_2(\hat{x}) - f_2(x)] + \mu_j p(x)$, one gets $f_2(\bar{x}) \ll f_2(\hat{x})$ and $\bar{x}$ is nearly feasible. Figure 3 shows this tunneling effect in the case of $p \equiv 0$ and $x \in \mathbb{R}$.

It is obvious that, in general, a solution of problem (6) is not a global solution of problem (4). But such a solution can be used as a new starting point (for instance) for a penalty method with the subproblem (5). The idea of this hybrid technique is based on the fact that descent methods in unconstrained optimization (like the BFGS method) determine points in the same valley where the starting point is located. A new starting point obtained by finding a tunnel to another valley may lead to a local solution with a smaller function value.

In order to solve the constrained optimization problem (6), we use a special penalty method with subproblems of the type

$$\min_{x \in \mathbb{R}^n} 1/[f_2(\hat{x}) - f_2(x)] + \mu_j p(x) + \nu_l \max\{0, f_2(x) - f_2(\hat{x}) + \epsilon\}^2, \quad (7)$$

where $(\nu_l)_{l \in \mathbb{N}}$ is a sequence of positive real numbers tending to infinity.

2.3. Interactive Procedure. Next, we summarize our preceding investigations and formulate an interactive method for the numerical solution of the bicriterial nonlinear vector optimization problem (1) under Assumption 1.1 with $\hat{S} := \mathbb{R}^n$.

The resulting algorithm reads as follows.

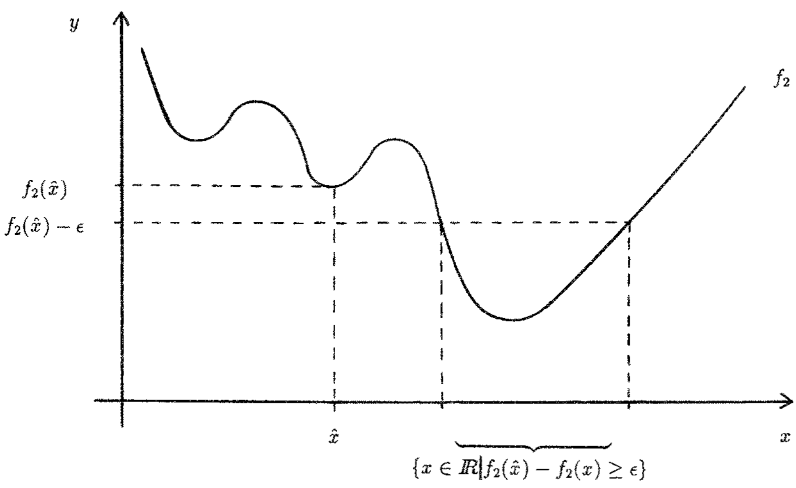

Fig. 3. Simplified illustration of the tunneling technique.

Algorithm 2.1.

Part 1. Computation Phase.

Step 1. Determine the real numbers a and b, where

$$a := \min_{x \in S} f_1(x),$$

$$b := f_1(\tilde{x}) \qquad \text{with} \quad f_2(\tilde{x}) = \min_{x \in S} f_2(x).$$

Step 2. For a given number $m \in \mathbb{N}$, determine the points

$$y_1^{(k)} := a + k(b-a)/m, \qquad k = 0, 1, \ldots, m.$$

Step 3. For every discretization point $y_1^{(k)}$, $k = 0, 1, \ldots, m$, determine a global solution $x^{(k)}$ of the constrained problem

$$f_2(x^{(k)}) = \min \quad f_2(x),$$
$$\text{s.t.} \quad x \in S,$$
$$f_1(x) = y_1^{(k)}.$$

For the determination of such a global solution, apply a penalty method with subproblems of type (5) solved with the BFGS method, compute a new starting point with a penalty method with subproblems of type (7) solved with the BFGS method, and again apply a penalty method with subproblems of type (5) solved with the BFGS method. This scheme may be carried out repeatedly, if the minimal values do not change.

Step 4. Select all minimal elements of the discrete set $\{f(x^{(0)}), f(x^{(1)}), \ldots, f(x^{(m)})\}$. Denote this set of minimal solutions by $\tilde{M}$.

Part 2. Decision Phase.

Step 5. The decision maker chooses the weights $t_1, t_2 > 0$ of the weighted Chebyshev norm in $\mathbb{R}^2$.

Step 6. The decision maker chooses an arbitrary reference point $y^{(1)} \in \mathbb{R}^2$.

Part 3. Computation Phase.

Step 7. Set $i := 1$.

Step 8. Determine a point $\bar{x}^{(i)} \in \tilde{M}$ with the property that

$$\max_{j=1,2}\{t_j|y_j^{(i)} - f_j(\bar{x}^{(i)})|\} \leq \max_{j=1,2}\{t_j|y_j^{(i)} - f_j(x)|\}, \qquad \text{for all } x \in \tilde{M}.$$

Part 4. Decision Phase.

Step 9. The point $\bar{x}^{(i)} \in \tilde{M}$ is presented to the decision maker. If the decision maker accepts this point as the subjectively best one, then stop. Otherwise, proceed to the next step.

Step 10. Based on additional information about the original problem using computational results obtained in Step 3, the decision maker proposes a new reference point $y^{(i+1)} \in \mathbb{R}^2$.

Part 5. Computation Phase.

Step 11. Set $i := i+1$, and go to Step 8.

Part 1 of Algorithm 2.1 is the computer-intensive part, while Parts 2–5 can be carried out interactively in a very fast way. So, Algorithm 2.1 is normally applied in such a way that Part 1 is executed independently from the other parts. The actual interactive method starts then with the elements of the set $\tilde{M}$.

Notice that it is not necessary to determine equidistant points $y_1^{(k)}$ in Step 2 of Algorithm 2.1. These points can also be chosen with the aid of other schemes as is done in the Polak method.

In Step 5, the decision maker can choose the weights of the weighted Chebyshev norm. This is important, because very often it does not make sense to compare the objectives numerically without scaling f_1 and f_2. In Step 10, it should be possible to provide the decision maker with all information obtained in the computation phases. For instance, the graphical presentation of the image set of minimal solutions enables the decision maker to select new reference points.

3. Numerical Results

The algorithm developed in the preceding section is now applied to two bicriterial nonlinear optimization problems. The first problem is very simple; it is designed for the illustration of the tunneling technique. The second problem is a complicated nonlinear structural optimization problem from mechanical engineering.

Example 3.1. We consider the following bicriterial optimization problem, taken from Ref. 8:

$$\min \begin{pmatrix} x_1 \\ x_2 \end{pmatrix},$$

$$\text{s.t.} \quad x_2 - 2.5 \leq 0,$$

$$(x_1 - 0.5)^2 - x_2 - 4.5 \leq 0,$$

$$-x_1 - x_2^2 \leq 0,$$

$$-(x_1 + 1)^2 - (x_2 + 3)^2 + 1 \leq 0,$$

$$(x_1, x_2) \in \mathbb{R}^2.$$

Since the objective mapping of this problem is the identity, the image and preimage sets are equal. This set is illustrated in Fig. 4. It can be seen from this figure that the set M of minimal elements is not connected (in fact, it consists of three arcs); therefore, a jump must be carried out between the disconnected parts. Such a jump can only be realized with the aid of a global minimization technique like the proposed tunneling technique. With a local minimization technique, it is in general not possible to jump from one arc of M to another one while increasing the $y_1^{(k)}$ values in the Polak method.

Table 1 lists the minimal elements $f(x^{(k)}) \in \tilde{M}$. These results were obtained with Part 1 of Algorithm 2.1. If one connects these computed points by straight lines where one notices that there are two jumps which

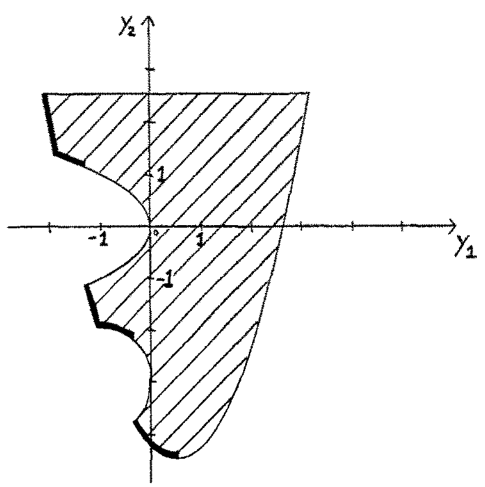

Fig. 4. Image set of the objective mapping.

Table 1. Iterated minimal elements ($y_1^{(k)}$'s chosen nonequidistantly).

k	$f(x^{(k)}) \in M$
0	(−2.131087, 2.422933)
1	(−1.930941, 1.409813)
2	(−1.731149, 1.315711)
3	(−1.531138, 1.238715)
4	(−1.331107, 1.156983)
5	(−1.131001, −1.207556)
6	(−0.931086, −2.002503)
7	(−0.731135, −2.036241)
8	(−0.526106, −2.117397)
9	(−0.315858, −3.729672)
10	(−0.115894, −4.013447)
11	(0.084136, −4.327440)
12	(0.500000, −4.500500)

can be located with the proposed tunneling technique, one gets a set illustrated in Fig. 5. This set is already a good approximation of the set M of minimal solutions.

If one chooses the weights $t_1 = t_2 = 1$ in Part 2 of Algorithm 2.1, one gets estimates as best approximations from the set of minimal elements given in Table 2.

Next, we investigate a structural optimization problem, namely the optimal design of a sandwich beam. This problem has already been formulated and solved by Eschenauer–Schäfer (e.g., see Refs. 9–11).

Example 3.2. We consider a sandwich beam consisting of a pitted aluminum core and covered by two aluminum coats. This beam is supported by five steel bars (see Fig. 6).

The design variables are as follows: x_1 = thickness of the coat; x_2 = height of the pitted core; x_3, x_4 = coordinates of the substructure; x_5 = diameter of bar No. 5; x_6 = diameter of bars No. 1 and No. 4; x_7 = diameter of bars No. 2 and No. 3.

The aim is to minimize the weight f_1 of the whole structure and the deformation f_2 of the beam under its net weight. These two objectives are given as follows:

$$f_1(x) = [2x_6\sqrt{x_3^2 + x_4^2} + 2x_7\sqrt{(l - 2x_3)^2/4 + x_4^2} + x_5(l - 2x_3)]\rho_s g f_s + (2x_1\rho_D + x_2\rho_k)b_s g f_s l, \tag{8a}$$

$$f_2(x) = \sqrt{(2/3)[\max\{\omega(\xi, x) | 0 \le \xi \le l/2\}]^2 + (1/3)\omega(l/2, x)^2}, \tag{8b}$$

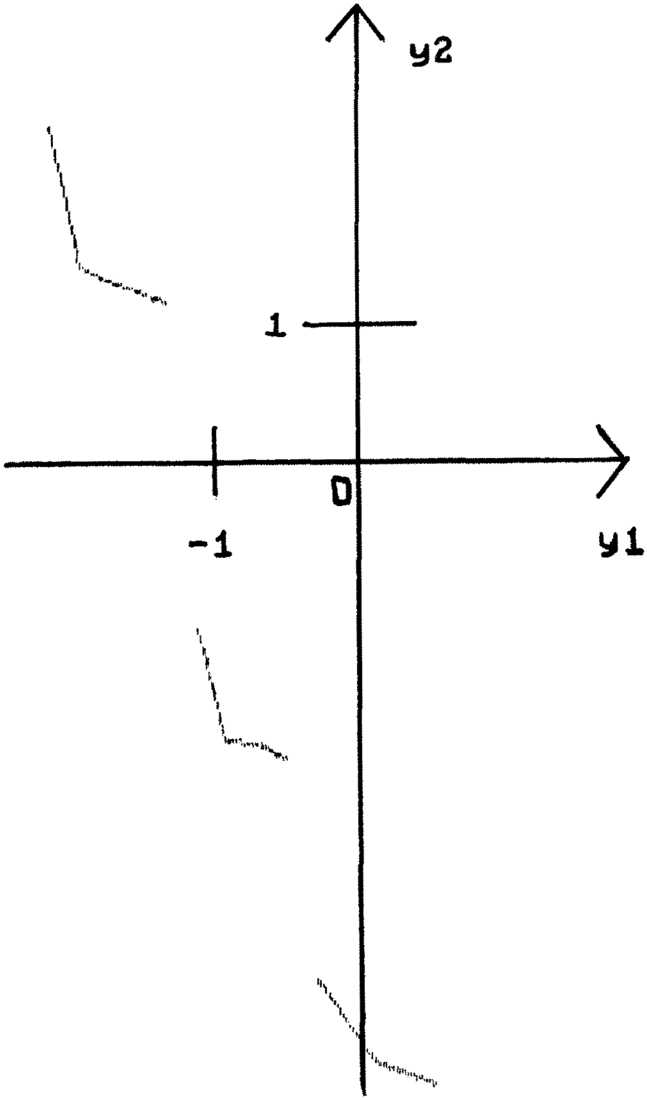

Fig. 5. Approximation of the set of minimal elements.

where

$$\omega(\xi, x) = [H_1(x)/6B_s(x)](\xi l^2 - \xi^3) + [q_0(x)/24B_s(x)](\xi^4 - \xi l^3) - [H_2(x)/48B_s(x)]\xi l^2 + [q_0(x)/2G_s(x)](\xi l - \xi^2) + [H_2(x)/2G_s(x)]\xi,$$

Table 2. Compromise solutions.

Reference point	Estimate as best approximation from the set of minimal elements
(−2, 0)	(−1.331107, 1.156983)
(0, 0)	(−1.131001, −1.207556)
(−1, 3)	(−2.131087, 2.422933)
(−2, 2)	(−2.131087, 2.422933)

with

$$H_1(x) = F_2(x)/2 + q_0(x)(l/2) + (2x_4/l)[F_4(x)x_3/x_4 - S_5(x)],$$

$$H_2(x) = F_2(x) + (4x_4/l)[F_4(x)x_3/x_4 - S_5(x)],$$

$$F_2(x) = x_7\sqrt{(l-2x_3)^2/4 + x_4^2}\rho_s g f_s,$$

$$F_4(x) = (1/2)[x_6\sqrt{x_3^2+x_4^2} + x_7\sqrt{(l-2x_3)^2/4+x_4^2} + x_5(l-2x_3)]\rho_s g f_s,$$

$$q_0(x) = (2x_1\rho_D + x_2\rho_k)b_s g f_s,$$

$$S_5(x) = [1/e(x)]\{F_2(x)[x_4 l/12B_s(x) + x_4/G_s(x)l]$$
$$- F_4(x)[(x_3/x_4)r + 4(x_3^2+x_4^2)^{3/2}/Ex_6x_4l^2 + 2x_3^2/E_DA_Dx_4l]$$
$$+ q_0(x)[5x_4l^2/96B_s(x) + x_4/2G_s(x)]\},$$

$$e(x) = (8/El^3)[(x_3^2+x_4^2)^{3/2}/x_6 + ((l-2x_3)^2/4+x_4^2)^{3/2}/x_7]$$
$$+ 4x_3^2/E_DA_Dl^2 + x_4^2/3B_s(x) + 4x_4^2/G_s(x)l^2 + (l-2x_3)/Ex_5l,$$

$$B_s(x) = E_Db_sx_1(x_1+x_2)^2/2,$$

$$G_s(x) = G_kb_s(x_1+x_2)^2/x_2.$$

The values of the constants are listed in Table 3.

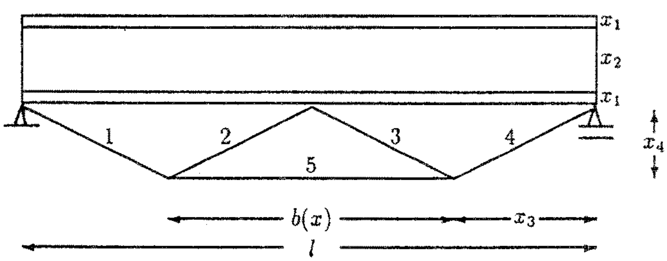

Fig. 6. Sandwich beam.

Table 3. Values of the constants.

Constant	Value
l	2000 mm
b_s	200 mm
ρ_D	2700 kg/m^3
E_D	70,000 N/mm^3
ρ_k	90 kg/m^3
G_k	30 N/mm^2
ρ_s	7860 kg/m^3
E_s	210,000 N/mm^2
g	9.81 m/s^2
f_s	1.2
$\sigma_{k\max}$	300 N/mm^2
s	1.4

Table 4. Iterated minimal elements ($y_1^{(k)}$'s chosen nonequidistantly).

$f_1(x^{(k)})$	$f_2(x^{(k)})$	$f_1(x^{(k)})$	$f_2(x^{(k)})$
8.170743	1613.325880	65.485503	0.032772
10.170946	335.297128	68.483001	0.031190
12.173212	129.391623	71.483001	0.029269
14.169546	63.646828	74.483001	0.027133
16.168230	37.121296	77.483001	0.024913
18.170628	22.107509	80.483001	0.022716
20.170631	15.165513	83.482979	0.020618
22.170864	11.005089	86.483001	0.018664
24.168286	8.316745	89.482979	0.016877
26.170787	6.480834	92.485484	0.015261
28.170658	5.187719	92.903848	0.015049
30.170622	4.241161	94.903869	0.014126
32.168918	3.534406	115.455054	0.011496
34.170188	2.988244	118.452682	0.009315
36.169843	2.558842	121.455039	0.008211
38.168907	2.215119	124.455107	0.007615
40.168608	1.935677	126.454850	0.007393
42.168488	1.705571	128.456033	0.007245
44.170746	1.160908	131.455268	0.007108
46.170746	1.006852	134.455584	0.007021
47.485498	0.271472	137.456063	0.006949
49.485507	0.109936	140.456706	0.006872
52.482961	0.054953	143.457513	0.006780
55.480575	0.041851	146.457594	0.006669
58.482930	0.037930	149.394510	0.006540
61.483003	0.034422	151.392035	0.006455
63.485505	0.033628		

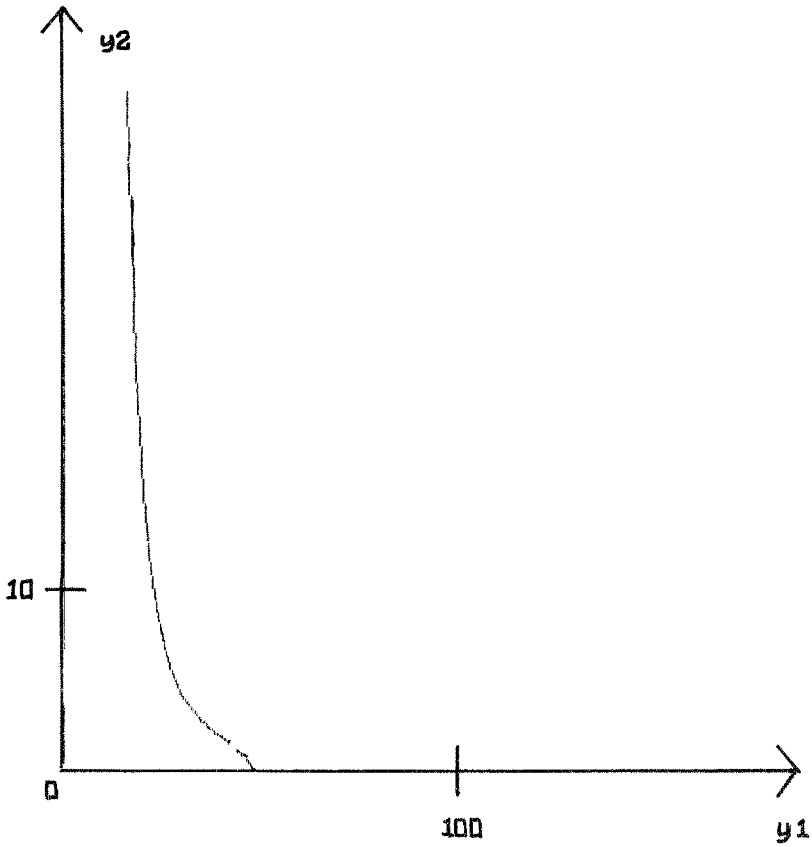

Fig. 7. Approximation of the set of minimal elements.

The constraints of this problem are given as

$$0.1 \le x_1 \le 2,$$

$$10 \le x_2 \le 100,$$

$$0 \le x_3 \le 990,$$

$$0 \le x_4 \le 990,$$

$$10 \le x_5 \le 100,$$

$$10 \le x_6 \le 100,$$

$$10 \le x_7 \le 100,$$

Table 5. Compromise solutions.

Reference point	Estimate as best approximation from the set of minimal elements	Preimage of the estimate
(50, 10)	(42.168488, 1.705571)	(1.408978, 11.848378, 989.930847, 0.248698, 10.000272, 10.049928, 10.000246)
(10, 10)	(18.170628, 22.107509)	(0.466426, 11.755359, 989.930632, 0.248085, 10.000682, 10.071396, 10.000504)
(30, 5)	(30.170622, 4.241161)	(0.937273, 11.840433, 989.930879, 0.248696, 10.000241, 10.046709, 10.000213)
(50, 2)	(49.485507, 0.109936)	(0.257025, 97.324846, 0.022438, 989.977789, 10.001214, 10.001056, 10.001630)

and

$$\begin{aligned} &|H_1(x)(l/2) - q_0(x)(l^2/8)|/(x_1+x_2)b_s x_1 \\ &+|-F_4(x)(x_3/x_4)+(2x_3/l)(F_4(x)x_3/x_4 - S_5(x))|/b_s x_1 \\ &\leq \sigma_{k\,\max}/s. \end{aligned} \tag{9}$$

So, we have to minimize the two functions f_1 and f_2 subject to these constraints. The application of Part 1 of Algorithm 2.1 leads to estimates of minimal elements of the image set of (f_1, f_2) listed in Table 4.

With the aid of the iterated minimal elements we obtain an approximation of the set of minimal elements illustrated in Fig. 7. If one chooses the weights $t_1 = t_2 = 1$ in Part 2 of Algorithm 2.1, one gets estimates as best approximations from the set of minimal elements given in Table 5.

We should notice that this structural optimization problem is in fact a nonsmooth problem [see (8) and (9)]. But nevertheless, it is possible to work with the BFGS method. It is well known that this method is not very sensitive for problems with mainly smooth functions. Of course, this statement is based on numerical experience and cannot be generalized.

4. Conclusions

It is shown in this paper that a reference point approximation method can be used successfully for the solution of real-world problems in bicriterial nonlinear optimization. The numerical effort in Part 1 of Algorithm 2.1 is immense, but the interactive phases can be carried out rapidly. From the point of view of the decision maker, it is important to note that only reference points are required which represent the decision maker's goal and which can be easily given. Moreover, it is important to mention that these reference points can be chosen arbitrarily and that they are not restricted to certain regions in the criteria space.

References

1. HWANG, C.-L., and MASUD, A. S. M., *Multiple Objective Decision Making—Methods and Applications*, Springer, Berlin, Germany, 1979.
2. JAHN, J., *A Method of Reference Point Approximation in Vector Optimization*, Operations Research Proceedings 1987, Edited by H. Schellhaas, P. Van Beek, H. Isermann, R. Schmidt, and M. Zijlstra, Springer, Berlin, Germany, pp. 576–587, 1988.
3. WIERZBICKI, A. P., *The Use of Reference Objectives in Multiobjective Optimization*, Multiple Criteria Decision Making—Theory and Application, Edited by G. Fandel and T. Gal, Springer, Berlin, Germany, pp. 468–486, 1980.
4. GRAUER, M., *Reference Point Optimization—The Nonlinear Case*, Essays and Surveys on Multiple Criteria Decision Making, Edited by P. Hansen, Springer, Berlin, Germany, pp. 126–135, 1983.
5. JAHN, J., *Parametric Approximation Problems Arising in Vector Optimization*, Journal of Optimization Theory and Applications, Vol. 54, pp. 503–516, 1987.
6. POLAK, E., *On the Approximation of Solutions to Multiple Criteria Decision Making Problems*, Multiple Criteria Decision Making, Kyoto 1975, Edited by M. Zeleny, Springer, Berlin, Germany, pp. 271–281, 1976.
7. LUENBERGER, D. G., *Linear and Nonlinear Programming*, Addison-Wesley, Reading, Massachusetts, 1984.
8. MERKEL, A., *Ein Verfahren der Referenzpunktapproximation für bikriterielle nichtlineare Optimierungsprobleme*, Diploma Thesis, University of Erlangen-Nürnberg, 1989.
9. ESCHENAUER, H., and SCHÄFER, E., *Interaktive Vektoroptimierung in der Strukturmechanik*, Zeitschrift für Angewandte Mathematik und Mechanik, Vol. 69, pp. T87–T89, 1989.
10. ESCHENAUER, H., and SCHÄFER, E., *Sandwichbalken mit Fachwerkunterbau*, Report, Forschungslaboratorium für Angewandte Strukturoptimierung, University of Siegen, Siegen, Germany, 1989.
11. ESCHENAUER, H., and SCHÄFER, E., *Unterprogramm STRAN zur Berechnung eines Sandwichbalkens*, Report, Forschungslaboratorium für Angewandte Strukturoptimierung, University of Siegen, Siegen, Germany, 1989.