Annals of Operations Research 132, 109–134, 2004

An Efficient Sampling Approach to Multiobjective Optimization

YAN FU yfu4@ford.com
Research and Advance Engineering Laboratories, Ford Motor Company, Dearborn, MI 48124, USA

URMILA M. DIWEKAR * urmila@uic.edu
CUSTOM, Center for Uncertain Systems: Tools for Optimization & Management, Departments of Bio, Chemical, and Industrial Engineering and Institute of Environmental Science & Policy, University of Illinois at Chicago, Chicago, IL 60607, USA

Abstract. This paper presents a new approach to multiobjective optimization based on the principles of probabilistic uncertainty analysis. At the core of this approach is an efficient nonlinear multiobjective optimization algorithm, Minimizing Number of Single Objective Optimization Problems (MINSOOP), to generate a true representation of the whole Pareto surface. Results show that the computational savings of this new algorithm versus the traditional constraint method increase dramatically when the number of objectives increases. A real world case study of multiobjective optimal design of a best available control technology for Nitrogen Oxides (NOx) and Sulfur Oxides (SOx) reduction illustrates the usefulness of this approach.

Keywords: nonlinear multiobjective optimization, MINSOOP algorithm, Hammersley sequence sampling, LTO process, NOx reduction

1. Introduction

Multiobjective problems appear in virtually every field and in a wide variety of contexts. The importance of multiobjective optimization can be seen by the large number of applications presented in the literature (Sobol, 1992; Schy and Giesy, 1988; Ohkubo, Dissanayake, and Taniwaki, 1998; Starkey, Gray, and Watts, 1988; Eschenauer, 1988; Silverman, Steuer, and Whisman, 1988; Tamiz and Jones, 1996; Olson, 1993; Kumar, Singh, and Tewari, 1991; NWTRB, 1996; Wood, Greis, and Steuer, 1982; Ferreira and Machado, 1996; Agrell, Lence, and Stam, 1998; Cohon, Scavone, and Solanki, 1988; Fu, 2000; Fu et al., 2000; Fu and Diwekar, 2003; Johnson and Diwekar, 2001). Most of these applications are multiobjective problems of nonlinear nature, which is why we need tools for nonlinear programming capable of handling multiple conflicting or incommensurable (e.g., different units) objectives.

* Corresponding author.

In this paper, we study a multiobjective optimization problem of the form:

$$\begin{aligned} \text{minimize} \quad & f_i(\vec{x}),\ i = 1, \ldots, k,\ k \geqslant 2, \\ \text{subject to} \quad & h_I(\vec{x}) = 0, I \geqslant 0, \\ & g_J(\vec{x}) \leqslant 0, J \geqslant 0, \\ & l_j \leqslant x_j \leqslant u_j,\ j = 1, \ldots, n, \\ & \vec{x} = (x_1, \ldots, x_n). \end{aligned} \tag{1}$$

The problem under consideration involves a set of n decision variables represented by the vector $\vec{x} = (x_1, \ldots, x_n)$. The equality constraints $h_I(\vec{x}) = 0, I \geqslant 0$: $\mathbb{R}^n \rightarrow \mathbb{R}$ and inequality constraints $g_J(\vec{x}) \leqslant 0$, $J \geqslant 0$: $\mathbb{R}^n \rightarrow \mathbb{R}$ are real-valued (possibly nonlinear) constraint functions, and l_j and u_j are the lower and upper bounds of the decision variable x_j (allowed to be $-\infty$ and/or $+\infty$). Both the equality constraints $h_I(\vec{x})$ and inequality constraints $g_J(\vec{x})$ are assumed to be continuously differentiable and the feasible decision region **S** defined by (1) is assumed to be a nonempty subset of $\mathbb{R}^n$. If $I = 0$ and $J = 0$, the problem becomes unconstrained. The decision situation involves k ($\geqslant 2$) continuously differentiable nonlinear objective functions $f_k : \mathbb{R}^n \rightarrow \mathbb{R}$. The vector of objective functions is $\mathbf{f}(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x}))^{\mathrm{T}}$ or $\mathbf{z} = (z_1, \ldots, z_k)^{\mathrm{T}}$, where $z_i = f_i(\vec{x})$ for all $i = 1, \ldots, k$ and the feasible objective region **Z** defined by (1) is assumed to be a nonempty subset of $\mathbb{R}^n$. Without the loss of generality, we assume that all the objective functions are to be minimized simultaneously (note that an objective of the maximization type could be converted to one of the minimization type by multiplying the objective function by -1).

As explained in numerous books and survey articles (Stadler, 1988; Osyczka, 1984; Yu, 1985; Cohon, 1978; Hwang and Masud, 1979; Steuer, 1986; Zeleny, 1974; Zoints, 1989; Stewart, 1992; Chankong and Haimes, 1983a, 1983b; Chankong et al., 1985; Evans, 1984; Rosenthal, 1985), it is not possible to find a single solution that would be optimal for all the objectives simultaneously because of the contradiction and possible incommensurability of the objective functions. In fact, the solution of a multiobjective optimization problem is a set of solution alternatives called the Pareto set. For each of these solution alternatives, it is impossible to improve one objective without sacrificing the value of another relative to some other solution alternatives in the set. A more formal definition of Pareto optimality is the following:

A decision vector $\vec{x}^* \in \mathbf{S}$ is Pareto optimal (also called Edgeworth–Pareto optimal, the efficient solution, the nondominated, the noninferior, the functional efficient) for problem (1) if there does not exist another decision vector $\vec{x} \in \mathbf{S}$ such that $f_i(\vec{x}) \leqslant f_i(\vec{x}^*)$ for all $i = 1, \ldots, k$ and $f_j(\vec{x}) < f_j(\vec{x}^*)$ for at least one index j.

An objective vector $\mathbf{z}^* \in \mathbf{Z}$ is Pareto optimal if there does not exist another objective vector $\mathbf{z} \in \mathbf{Z}$ such that $z_i \leqslant z_i^*$ for all $i = 1, \ldots, k$ and $z_i < z_i^*$ for at least one index j; or equivalently, $\mathbf{z}^*$ is Pareto optimal if the decision vector corresponding to it is Pareto optimal. There are usually many (infinite in number) Pareto optimal solutions. The collection of these is called the Pareto set. It is from this subset of potential solutions that the final, preferred decision is chosen by the decision-makers.

There is a large array of analytical techniques to solve a multiobjective optimization (programming) problem; however, the MOP methods are generally divided into two basic types: preference-based methods and generating methods. Preference-based methods like goal programming attempt to quantify the decision-maker's preference, and with this information, the solution that best satisfies the decision-makers's preference is then identified (Diwekar, 2003). Generating methods, such as the weighting method and the constraint method, have been developed to find the exact Pareto set or an approximation of it. In this paper, we are concentrating on the constraint method.

The basic strategy in constraint methods (Haimes, Lasdon, and Wismer, 1971; Cohon, 1978; Zeleny, 1982; Diwekar, 2003) is to transform the multiobjective optimization problem into a series of single objective optimization problems. The idea is to pick one of the objectives to minimize (say Z_l) while each of the others ($Z_i, i = 1, \ldots, k, i \neq l$) is turned into an inequality constraint with parametric right-hand sides ($\varepsilon_i, i = 1, \ldots, k, i \neq l$). The problem takes the form:

$$\begin{aligned} \text{minimize } & Z_l = f_l(\vec{x}) \\ \text{subject to } & Z_i = f_i(\vec{x}) \leqslant \varepsilon_i, \ i = 1, \ldots, k, \ k \neq l, \\ & h_I(\vec{x}) = 0, \ I \geqslant 0, \\ & g_J(\vec{x}) \leqslant 0, \ J \geqslant 0, \\ & l_j \leqslant x_j \leqslant u_j, \ j = 1, \ldots, n, \\ & \vec{x} = (x_1, \ldots, x_n). \end{aligned} \tag{2}$$

Solving repeatedly for different values of $\varepsilon_i, \ldots, \varepsilon_{l-1}, \varepsilon_{l+1}, \ldots, \varepsilon_k$ leads to the Pareto set. This method also needs to obtain solutions for a large number of single objective optimization problems. Some of the theoretical results of the constraint method (Miettinen, 1999) are (a) a decision vector $\mathbf{x}^* \in \mathbf{S}$ is Pareto optimal if and only if it is a solution of the constraint problem (2) for every $l = 1, \ldots, k$, where $\varepsilon_i = f_i(\mathbf{x}^*)$ for $i = 1, \ldots, k, i \neq l$, (b) a point $\mathbf{x}^* \in \mathbf{S}$ is Pareto optimal if it is a unique solution of the constraint problem for some l with $\varepsilon_i = f_i(\mathbf{x}^*)$ for $i = 1, \ldots, k, i \neq l$, and (c) the unique solution of the constraint problem (2) is Pareto optimal for any given upper bound vector $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_{l-1}, \varepsilon_{l+1}, \ldots, \varepsilon_k)^T$.

The computational burden of the constraint methods is more laborious than the weighting methods for finding each Pareto solution because the number of constraints is larger than that of the weighting method, and sometimes no feasible solution can be found for some particular combinations of the right-hand sides. In reality, the influence is assumed to be trivial since there are often a much larger number of original constraints than the number of objectives in large-scale real-world applications. Furthermore, the constraint method offers the advantages of better control over exploration of the Pareto set and of being able to locate points anywhere along the Pareto surface. Theoretically, every Pareto optimal solution of any multiobjective optimization problem can be found by the constraint method through altering the upper bounds and the function to be minimized (Miettinen, 1999). Therefore, in this paper, an efficient multiobjective optimization is developed and rooted in the constraint method to solve large-scale real-world problems.

Ideally, a method that can generate the whole Pareto set is the most desirable one. However, the inconvenience here is that the generating process is usually expensive and sometimes difficult, if not impossible because of the large number of outcomes. Further, it is difficult to judge the completeness of the Pareto set for higher dimensions. In practice, as suggested by Benson and Sayin (1997), instead of trying to generate the whole Pareto set, one should aim to find a truly global representation of it. Probabilistic uncertainty analysis methods deal with large amount of data, errors, as well as large-scale multi-dimensional problems like molecular simulations and dynamics (Kim and Diwekar, 2002; Sahin and Diwekar, 2002). Therefore, these methods can be used for representation of the large amount of data encountered in MOP problems. One of the goals of this paper is to develop an efficient nonlinear multiobjective optimization algorithm based on efficient sampling methods used in uncertainty analysis, to provide an accurate representation of the whole Pareto set and to compare the performance of the new algorithm developed in this paper with the traditional constraint method under various conditions.

The rest of the paper is organized as follows. Section 2 introduces the probabilistic uncertainty analysis approach to MOP. This section presents a new nonlinear multiobjective optimization algorithm based on the Hammersley sequence sampling technique is introduced. In section 3, computational tests of the new algorithm and the traditional constraint method are conducted with different nonlinear convex multiobjective optimization problems under various conditions, and the results are compared. In section 4, the new algorithm is extended to a class of nonlinear nonconvex problems. Also presented in this study is a large-scale real world case study of designing a novel NOx control process that also reduces SOx and Carbon Dioxide (CO_2). The overall conclusions are presented in section 5.

2. Uncertainty analysis and the new approach to MOP

This section begins with the description of methods used to represent the complete Pareto set by a typical multiobjective linear problem. Then the new criteria derived from probabilistic uncertainty analysis are introduced to judge and compare different generating techniques. The section continues with discussions of various sampling techniques for uncertainty analysis and the uniformity property of a new sampling technique, followed by the introduction of a new nonlinear multiobjective optimization algorithm. The basic programming structure for the new algorithm is also provided at the end of this section.

2.1. Representation

There are few methods that have been published which specifically address the challenge of representation of multiobjective optimization Pareto surface with many objectives (Armann, 1989; Benson and Sayin, 1997; Sayin, 2000, 2003). Problems with two or, perhaps, three objectives permit a clear presentation of the Pareto set through graphical means. If there are more than three objectives, it is no longer possible to see the Pareto set graphically. An issue to consider is how to represent the Pareto set so that we can compare different generating methods in a systematical and quantitative way. In the recent papers Sayin (2000, 2003) addressed this question by presenting a mixed integer problem for calculation of coverage error. However, her formulation is limited to multiobjective linear programming problems. Another difficulty with the calculation of coverage error is that the mixed integer optimization problem size grows considerably with number of objectives. In this work, we are addressing this problem by using probabilistic uncertainty analysis method used for engineering systems.

The probabilistic uncertainty analysis method involves four steps: (1) assigning probability distributions to the key input variables, (2) sampling these probability distributions using efficient sampling techniques, and (3) propagating each sample through the model, and (4) analyzing the probabilistic output data (Diwekar and Rubin, 1991). The first two steps are linked to the representation of input uncertainties. To accommodate the diverse nature of uncertainties various distribution functions are used. These distributions provide a way of handling little to large amount of data representation. Characterization of these distributions involves collapsing the available data in terms of type of distribution, and the moments of the distribution. Usually, a distribution is quantified using the first two moments, namely, the mean and the variance. Occasionally other moments are also considered. If there are correlated uncertainties, correlation structure is used while sampling the distributions of uncertain parameters.

In this paper, we use the same notion of representing the distribution of solutions in a Pareto surface using the two moments. The mean and variance of an approximate Pareto surface are used as generalized representations of the whole Pareto set and as criteria to measure the accuracy of a generating method. It should be noted that in this surface the objective functions are correlated through decision variables. Therefore, separate correction check is not required. We adopt the following decision rules for determining the performance of generating methods:

(1) We estimate the “true” mean and variance of the whole Pareto surface for each problem by obtaining a very large number of Pareto optimal solutions (approximately 5000–10,000 subproblems) covering the whole Pareto set.

(2) Once the “true” mean and variance are established, the efficiency of different generating methods is measured by estimating the computational time needed to settle to within the same accuracy of the “true” mean and variance values. For the same type of optimality-based methods, the number of single objective optimization problems (or sub-problems) to be solved can be used as an approximation of the computational time required. Therefore, the number of sub-problems to be solved to the same accuracy of the mean and variance of an approximate Pareto surface is used to compare the efficiency of the new method to the traditional constraint method.

Table 1
Decision values and objective values for the extreme points for example 1.

	x_1	x_2	Z_1	Z_2
A	0	0	0	0
B	6	0	−30	6
C	6	2	−26	−2
D	4	4	−12	−12
E	1	4	3	−15
F	0	3	6	−12

The more accurate the mean and variance of an approximate Pareto surface to the "true" mean and variance of the Pareto surface, the closer the approximate Pareto set is to the Pareto set. This concept can be easily understood with example 1, problem (3), borrowed from Cohon (1978) with the difference of changing from a multiobjective maximization problem to a minimization one.

Example 1 (2 decision variables, 4 constraints, 2 objectives, multiobjective linear programming (MOLP)).

$$\begin{aligned}\text{Minimize } & Z_1 = -5x_1 + 2x_2,\\ & Z_2 = x_1 - 4x_2,\\ \text{subject to } & -x_1 + x_2 \leqslant 3,\\ & x_1 \leqslant 6,\\ & x_1 + x_2 \leqslant 8,\\ & x_2 \leqslant 4,\\ & x_1, x_2 \geqslant 0.\end{aligned} \tag{3}$$

Example 1 is a multiobjective linear problem (MOLP) for which we have two objectives, two decision variables, and four constraints. Since it is a simple multiobjective linear problem, the extreme points are obtained in table 1, and the feasible objective space is shown in figure 1 with the bold line BCDE as the Pareto set.

Figure 2 shows the differences between two approximate Pareto sets and the Pareto set when the numbers of sub-problems solved are 5 and 50, respectively. When the number of sub-problems solved is 5, there is a 10% relative error of the mean and a 150% relative error of the variance of the approximate Pareto set from the true mean and variance of the Pareto set. However, when the number of sub-problems solved is 50, the relative errors of the mean and variance are reduced to 0.7% and 9.5%, respectively. From figure 2, it is obvious that the more accurate the method is in estimating the mean and variance of the Pareto surface, the better it is at representing the Pareto set and vice versa. Therefore, in this paper the mean and variance of an approximate Pareto surface are calculated as generalized representations of the whole Pareto set for generating methods.

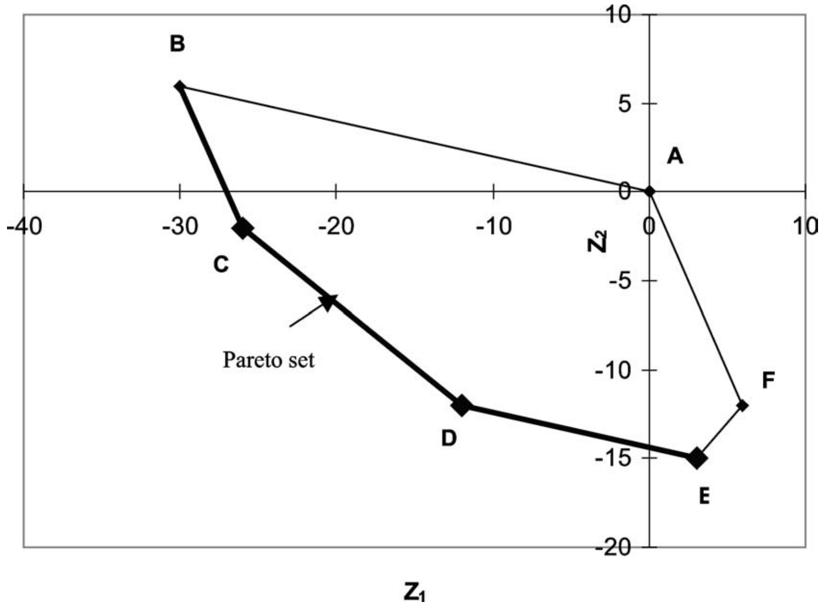


Figure 1. Objective space for example 1.

2.2. Criteria

Generating Pareto optimal solutions plays an important role in multiobjective optimization, and mathematically, the problem is considered solved when the Pareto set is found for a generating method. However, the actual solutions are perhaps infinite in number. Therefore, instead of finding the complete Pareto set, in practice it is often sufficient to find a true representation of the Pareto set. The more realistic target is to approximate discrete set of Pareto-optimal points (Sayin, 2000, 2003; Lampinen, 2000). Here are the five criteria we found to judge and compare different generating methods:

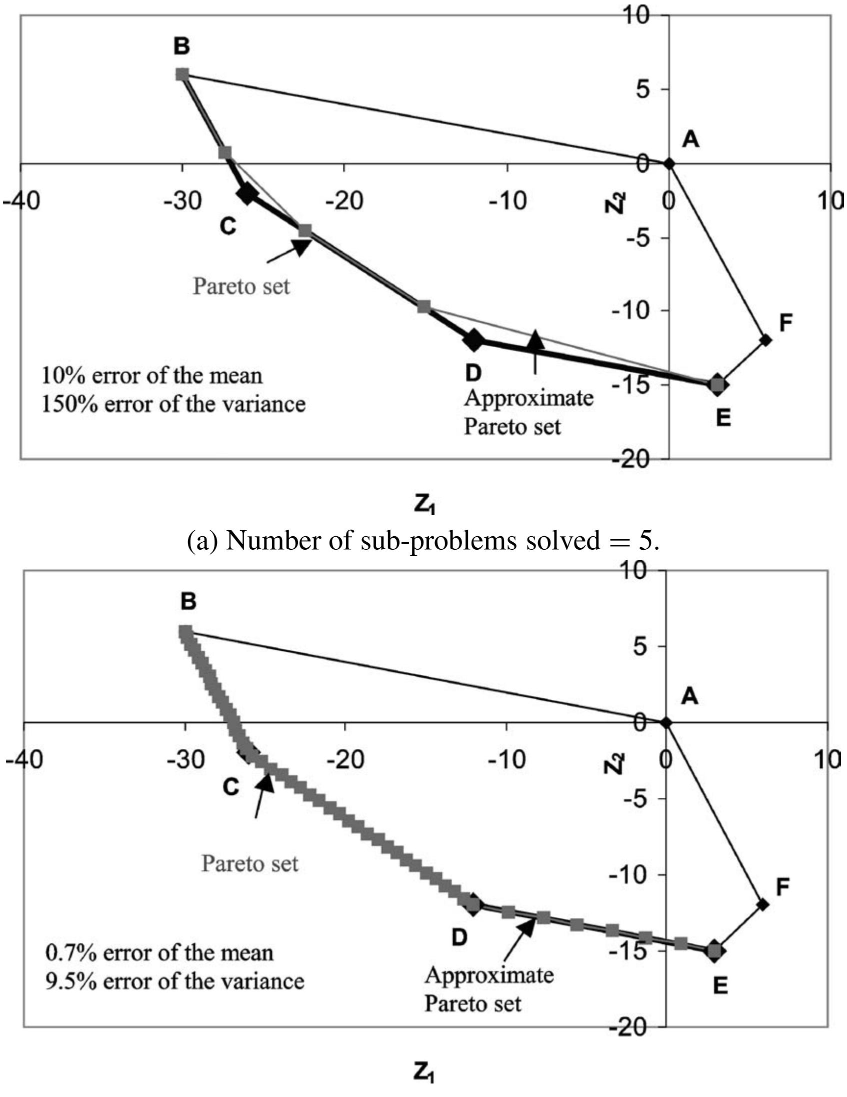
(1) *Completeness*. It can cover the whole Pareto set. For example, in the objective space of example 1 as shown in figure 1, the bold line BCDE is the Pareto set, and the completeness in this case means that the Pareto optimal solutions obtained by generating methods should cover the whole line BCDE, not just a segment of it.

(2) *Accuracy* is defined in terms of closeness to the mean and variance of the true Pareto surface.

(3) *Computational efficiency*. It can obtain an approximate Pareto set with the required accuracy by solving a minimum number of single objective optimization problems for optimality-based methods.

(4) *Robustness*. It can solve different types of problems, like linear, nonlinear convex and nonconvex multiobjective problems.

(5) *Automation*. It can automatically formulate and solve a family of single objective optimization problems, and can generate a representation of the Pareto set.



(a) Number of sub-problems solved = 5.

(b) Number of sub-problems solved = 50.

Figure 2. The difference between two approximate Pareto sets and the Pareto set with different numbers of sub-problems solved for example 1.

2.3. *Hammersley sequence sampling technique*

As discussed in section 1, we think the constraint method is a better candidate as the basis to develop an efficient new algorithm because it is more robust than the weighting method. Therefore the focus of this paper is to develop a new multiobjective nonlinear programming (MONLP) algorithm that is based on the traditional constraint method, but can obtain significant computational savings as compared to the current method.

Recall that the constraint method requires us to pick one of the objectives to minimize (e.g., Z_l), while each of the others (Z_i, $i = 1, \dots, k$, $i \neq l$) is turned into an inequality constraint with a parametric right-hand side (ε_i, $i = 1, \dots, k$, $i \neq l$). By solving repeatedly for different values of $\varepsilon_i, \dots, \varepsilon_{l-1}, \varepsilon_{l+1}, \dots, \varepsilon_k$, an approximation of the whole Pareto set is obtained. This involves solving a large number of optimization problems, as described below.

The algorithm is based on the assumption that the problem is equivalent to the results of calculating an integral over the space of objectives. Consider the approximation of an integral of a $(k - 1)$-dimensional (with a k-objective problem) continuous function by sampling its values at a finite set of points. One straightforward approach is to place the points along equally spaced intervals on a $(k-1)$-dimensional grid, which represents the traditional constraint method. Although this is a good arrangement, the number of points required increases rapidly as the number of objectives increases. For example, if there are six objectives and five of them are evaluated over 10 points for each objective, we would have to solve 100,000 optimization problems. Alternatively, one can use a Monte Carlo sampling (MCS) technique, where the points are chosen randomly. The approximation of the integral is then based on the function evaluation at these points. On average, however, the error of approximation (from central limit theorem) is of the order $O(N^{-1/2})$, which also means the number of points (N) required to keep the error within ε is bounded by $1/\varepsilon^2$ (Diwekar, 2003). The remarkable feature is that the bound is not dependent on the dimension ($k - 1$ in this case). This means MCS methods are unlikely to scale exponentially with increasing objective functions. However, MCS methods are based on pseudo-random number generators, and do not have good uniformity.

Recently, Kalagnanam and Diwekar (1997) developed an efficient sampling technique called the Hammersley sequence sampling (HSS) technique based on a quasi-random number generator. It uses the Hammersley points ((Hammersley, 1960), see appendix A) to uniformly sample a $(k-1)$-dimensional hypercube, and the results revealed that the Hammersley points provide the optimal location for the sample points so as to obtain better uniformity in the $(k - 1)$-dimension. The main reason for this is that the Hammersley sequence is a low-discrepancy design for placing n points on a $(k - 1)$-dimensional hypercube. In contrast, other stratified techniques such as the Latin hypercube are designed for uniformity along a single dimension and then randomly paired for placement on a $(k-1)$-dimensional cube. Therefore, the likelihood of such schemes providing good uniformity properties on high-dimensional cubes is extremely small. One of the main advantages of Monte Carlo methods over that of a uniform grid is that the number of samples required to obtain given accuracy of estimates does not scale exponentially with number of uncertain variables. HSS preserves this property of Monte Carlo (Kim and Diwekar, 2003). The number of points required to converge to the mean and variance of the derived distributions by the HSS method is on average 3 to 100 times less than the MCS and other stratified sampling techniques (Kalagnanam and Diwekar, 1997).

Table 2
Payoff table for example 2 (4 decision variables, 3 constraints, 3 objectives).

	Z_1	Z_2	Z_3
min Z_1	930.863	769.621	**1406.023**
min Z_2	1130.76	651.794	1386.973
min Z_3	**1161.44**	**783.55**	1316.853

2.4. A new MONLP algorithm – MINSOOP

The new multiobjective optimization algorithm, MInimizing Number of Single Objective Optimization Problems (MINSOOP), proposed in this paper uses the HSS technique to generate combinations of the right-hand sides ε_i ($i = 1, \ldots, k$, $i \neq l$) of the traditional constraint method. The steps for a multiobjective problem with k objectives (to be minimized) are listed as follows:

Step 1. Solve k single objective optimization problems individually with the original constraints of a multiobjective problem to find the optimal solution for each of the individual k objectives.

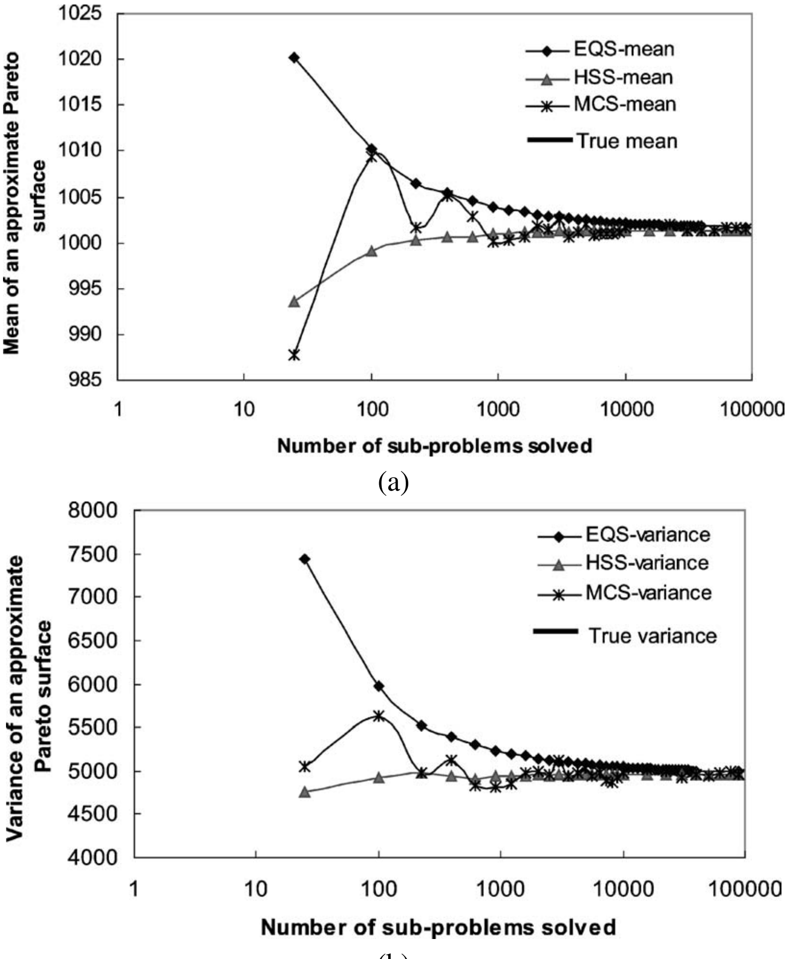
Step 2. Compute the value of each of the k objectives at each of the k individual optimal solutions. In this way, an approximation of the potential range of values for each of the k objectives is determined and saved in a table (called payoff table). The minimum possible value is the individual optimal (minimizing) solution. The approximate maximum possible value of the Pareto set is the maximum value for that objective found when minimizing the other $k - 1$ objectives individually.

Step 3. Select a single objective (e.g., Z_l) to be minimized. Transform the remaining $k - 1$ objectives into inequality constraints of the form $Z_i \leqslant \varepsilon_i$, $i = 1, \ldots, k$, $i \neq l$ and add these new $k - 1$ constraints to the original set of constraints. Then the original multiobjective optimization problem is transformed into a family of single objective optimization problems with parametric right-hand sides.

Step 4. Select a desired number of single objective optimization problems to be solved to represent the Pareto set. Using the HSS technique to generate the desired number of combinations of the inequality constraint values $\varepsilon_i, \ldots, \varepsilon_{l-1}$, $\varepsilon_{l+1}, \ldots, \varepsilon_k$ within the range determined in step 2.

Step 5. Solve the constrained problems set up in step 4 for every combination of the right-hand side values determined in step 3. These feasible solutions form an approximation for the Pareto set.

Consider the nonlinear convex multiobjective problem shown in example 2, with 4 decision variables, 3 constraints and objectives. The upper and lower bounds of the objectives are shown in the payoff table in table 2.



(a)

(b)

Figure 3. The performances of different methods for example 2.

Example 2 (4 decision variables, 3 constraints, 3 objectives, MONLP).

$$
\begin{aligned}
\text{Minimize } & Z_1 = (x_1 - 8)^2 + (x_2 - 12)^2 + (x_3 - 30)^2 + (x_4 - 10)^2, \\
& Z_2 = (x_1 - 10)^2 + (x_2 - 7)^2 + (x_3 - 8)^2 + (x_4 - 25)^2, \\
& Z_3 = (x_1 - 35)^2 + (x_2 - 10)^2 + (x_3 - 12)^2 + (x_4 - 7)^2, \\
\text{subject to } & \frac{x_1}{3} + \frac{x_2}{10} + \frac{x_3}{7} + \frac{x_4}{8} \leqslant 1, \\
& \frac{x_1}{15} + \frac{x_2}{12} + \frac{x_3}{5} + \frac{x_4}{10} \leqslant 1, \\
& \frac{x_1}{10} + \frac{x_2}{12} + \frac{x_3}{8} + \frac{x_4}{4} \leqslant 1, \\
& x_1, x_2, x_3, x_4 \geqslant 0.
\end{aligned}
\tag{4}
$$

Figure 3 shows the mean and variance of an approximate Pareto surface versus the number of single objective optimization problems (sub-problems) solved by different methods for a nonlinear convex multiobjective problem (4) (example 2), where the right-hand sides are generated by the equal space code (EQS, represents the traditional constraint method based on uniform grid), the Monte Carlo sampling (MCS, pseudo-random number generator), and the Hammersley sequence sampling (HSS). Results show that the method based on the HSS technique generally requires solving a fewer number of single objective optimization problems than with both the traditional constraint method and the method based on the MCS technique to converge to the "true" mean and variance of the Pareto surface. This illustrates that the new algorithm called MINSOOP can provide significant computational savings compared to the traditional constraint method in obtaining an accurate representation of the Pareto set.

Theoretically, every Pareto optimal solution of any multiobjective optimization problem can be found by the constraint method by altering the upper bounds and the function to be minimized. To ensure that a solution produced by the constraint method is Pareto optimal, we need to either solve k different problems or obtain a unique solution (Miettinen, 1999). In general, the uniqueness is not easy to verify. However, if the problem is convex and the function f_l to be minimized is strictly convex, we know that the solution is unique without further checking (Chankong and Haimes, 1983b). For the sake of simplicity, we use all nonlinear convex examples to compare the performance of the new algorithm with the traditional constraint method in section 3. However, a nonlinear nonconvex example is also provided in section 4 to demonstrate that the new method is able to generate a representation of the whole Pareto set for nonlinear nonconvex problems. Further improvement of a nonlinear optimizer to overcome the local minimum problem is discussed in section 4 as well so that a representation of the global Pareto set, instead of the locally Pareto set of a nonlinear nonconvex problem will be obtained.

3. Computational tests

In this section, we present the results of our computational tests for comparing the MINSOOP algorithm developed in this paper with the traditional constraint method.

3.1. Tests design

For the sake of simplicity, all examples used here have nonlinear convex objective functions and linear constraints. The general form of the examples is described in (5), and there are k objectives to be minimized, m inequality constraints, and n decision variables.

$$\begin{aligned} &\text{Minimize } Z_i = \sum_{j=1}^{m}(x_j - a_{ij})^2,\ i = 1,\ldots,k,\ k \geqslant 2, \\ &\text{subject to} \quad \sum_{l=1}^{m} \frac{x_j}{b_{lj}} \leqslant 1,\ m \geqslant 0, \\ &\qquad\qquad\quad x_j \geqslant 0,\ j = 1,\ldots,n. \end{aligned} \tag{5}$$

As proved by Yu (1985), the Pareto set for the above kind of problems is convex, and according to Chankong and Haimes (1983b) we know that every solution of each single objective optimization problem is Pareto optimal without further checking.

Next, results from a large number of numerical tests are presented. The design of the tests included varying the number of sub-problems solved, the numbers of decision variables, constraints and objectives involved, and comparing results with different accuracies of approximate Pareto sets representations. The details of the test design are as follows:

- *MONLP methods*. The traditional constraint method and the new MINSOOP algorithm are compared.
- *Number of sub-problems*. The number of sub-problems solved is varied from 10 to 100,000 for each case study.
- *Number of decision variables*. The number of decision variables ranges from 2 to 5, with the fixed 3 constraints and 3 objectives.
- *Number of constraints*. The number of constraints ranges from 2 to 5, with the fixed 4 decision variables and 3 objectives.
- *Number of objectives*. The number of objectives ranges from 2 to 5, with the fixed 4 decision variables and 3 constraints.
- *Accuracy*. The accuracy of the mean of approximate Pareto surfaces is varied from 99% to 99.9%, and the accuracy of the variance of approximate Pareto surfaces is changed from 80% to 99%.

3.2. Efficiency of the MINSOOP algorithm

Ten nonlinear convex multiobjective examples are studied here, each corresponding to different numbers of decision variables, constraints, and objectives. Similar results, as shown in figure 3, are also found for each of these problems. Due to the length of this paper, the complete figures of the mean and variance of an approximate Pareto surface versus the number of sub-problems solved by the traditional constraint method and the new MINSOOP algorithm for these problems (Fu, 2000) is not repeated here. In general, the larger the number of sub-problems solved, the closer the mean and variance of an approximate Pareto surface is to the "true" mean and variance of the Pareto surface. When the number of sub-problems is very large, both the new and traditional methods converge to the same "true" mean and variance of the Pareto surface. However, MINSOOP requires solving a fewer number of single objective optimization problems in order to converge to the "true" mean and variance of the Pareto surface as compared to the traditional method. This illustrates that the MINSOOP algorithm can offer significant computational savings and make it practical to use it for solving large-scale real-world problems.

Table 3
The efficiency of the new algorithm to the traditional constraint method for different numbers of decision variables.

Number of decision variables	2	3	4	5
$N_{\mathrm{EQS}}/N_{\mathrm{HSS}}$ for 99.9% of mean	163	8	14	9
$N_{\mathrm{EQS}}/N_{\mathrm{HSS}}$ for 99% of variance	3	20	225	40

Note. Where N denotes the number of sub-problems solved. The subscript EQS and HSS correspond to equal space and HSS methods.

3.3. Effect of number of decision variables

We changed the number of decision variables for the three constraints and three objectives representation of the problem (5). Results indicate that the number of nonlinear sub-problems needing to be solved for a fixed accuracy has no obvious relationship to the number of decision variables for both methods. However, the new algorithm (using HSS) usually solves for fewer sub-problems than the traditional method (using EQS), as indicated table 3.

3.4. Effect of number of constraints

When the number of constraints varies from two to five, it has been found that the number of nonlinear sub-problems needing to be solved for fixed accuracy is not sensitive to the number of constraints for both methods. However, the new algorithm usually requires fewer nonlinear sub-problems to be solved than the traditional method.

3.5. Effect of number of objective functions

Figure 4 shows the number of nonlinear sub-problems solved in an approximate Pareto surface versus the number of objectives using the same accuracy for both methods (e.g., 99.9% of the mean and 99% of the variance). Results show that the number of nonlinear sub-problems needing to be solved for the fixed accuracy increases rapidly as the number of objective functions increases for the traditional constraint method. However, there is no significant increase for the new algorithm as the number of objectives increases. This indicates that the computational savings by using the new nonlinear multiobjective optimization algorithm increases dramatically as the number of objectives increases. This result is particularly important, as the computational burden for large number of objectives is often extreme for all other multiobjective procedures.

3.6. Effect of accuracies

As discussed in the earlier studies, the number of nonlinear sub-problems needing to be solved by the new algorithm is generally less than the number needed with the traditional constraint method in order to obtain the same accuracy for the mean and variance of an approximate Pareto surface. In other words, by solving the same number of nonlinear

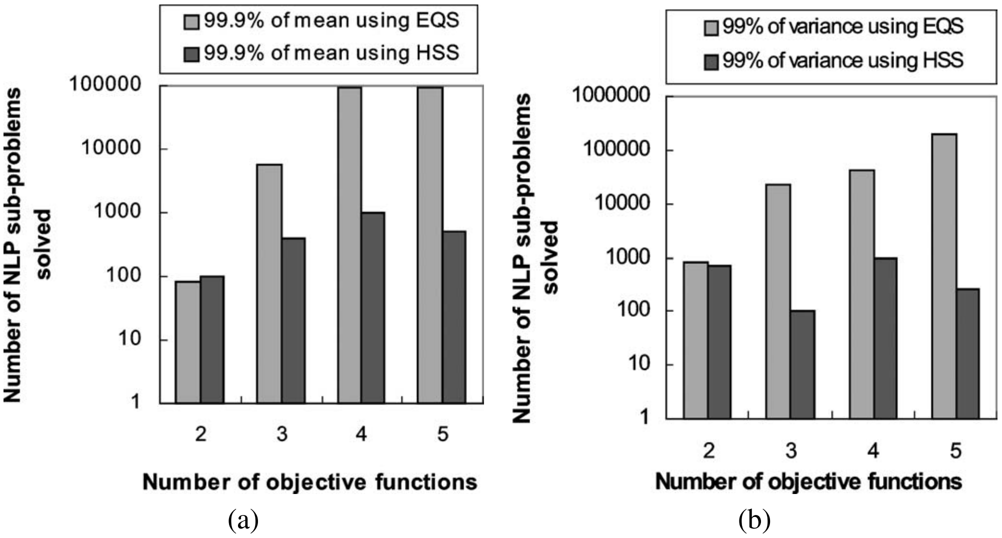


Figure 4. Compare the new algorithm and the existing constraint method with different numbers of objective functions.

sub-problems, the new algorithm usually attains a higher accuracy for the mean and variance of an approximate Pareto surface than can be obtained with the traditional method. Also, when the accuracies of the mean and variance of an approximate Pareto surface increase, the new algorithm is a better choice since the number of sub-problems to be solved by the traditional method increases rapidly as compared to the new algorithm. This means the new algorithm can provide even more significant computational savings as compared to the traditional constraint method when the accuracy demands are high.

4. Nonlinear nonconvex multiobjective problems

Theoretically, every Pareto optimal solution of any multiobjective optimization problem can be found by the constraint method by altering the upper bounds and the function to be minimized. It must be stressed that even duality gaps in nonconvex problems (see Miettinen (1999), Chankong and Haimes (1983b)) do not disturb the functioning of the constraint method. If a problem is convex and the function f_l to be minimized is strictly convex, then the solution produced by the constraint method is Pareto optimal without further checking. When the problem is nonconvex, we have to ensure that a solution is a unique solution. However, computationally, the uniqueness conditions are not always satisfied. In this section, a complete efficient nonlinear nonconvex example is used to demonstrate that the new algorithm inherits the robustness of the traditional constraint method, and then suggestions for improving a nonlinear optimizer to obtain globally Pareto optimal solutions for a general nonlinear nonconvex problem are discussed as well.

4.1. Real world case study

On September 24, 1998, new regulations announced by the US EPA require 22 Eastern States plus the District of Columbia to develop state implementation plans to reduce ground-level ozone through the reduction of nitrogen oxide (NOx) emissions (Cooper, 1998). This plan calls for a 28% NOx cut in the summer time (1.2 million tons) by 2007. This calls for utilities to develop new, efficient, and robust post-combustion NOx control technologies. A novel environmental control technology called Low Temperature Oxidation (LTO) system, which can reduce NOx emissions below measurable levels (2 ppm) using process analyzers at low temperature (125–325°F), was awarded Best Available Control Technology (BACT) and Lowest Available Emission Reduction technology (LAER) by the US EPA in April 1998. This technology also won the 2001 Kirkpatric award by chemical engineering magazine. In LTO, ozone is employed to oxidize nitric oxide (NO) to dinitrogen pentoxide (N_2O_5) at a low temperature in an oxidizer, which is then easily absorbed by water in a scrubber. Bench scale and pilot plant tests have shown that the LTO process can reduce the NOx emissions below the measurable levels using process analyzers (almost complete removal). This proved that the LTO system is an attractive process to meet the stricter NOx regulations. There are multiple benefits of the LTO system besides removal of NOx emissions, includes reduction of SOx and CO emissions, and no secondary air emissions (NH_3, N_2O). In order to obtain minimum NOx emissions, extra ozone needs to be supplied. The cost of the process also increases nonlinearly as emissions decrease. This poses a challenging multiobjective optimization problem where emissions like NOx and SOx need to be minimized, while minimizing the system cost as well as extra ozone. This problem is addressed here using the new and efficient multiobjective optimization MINSOOP algorithm.

Figure 5 shows the LTO process diagram for industrial boiler application. In figure 5, the exhaust gases from the boiler are cooled in a high temperature economizer, giving up heat to preheat the boiler feed water. The gas is then cooled to its dew-point temperature in the condensing economizer and a portion of the water vapor in the gas is condensed. Ozone is injected into the gas as it leaves the condensing economizer and is thoroughly mixed with the gas in a static mixer. The NOx in the gas is oxidized in the oxidation chamber to dinitrogen pentoxide (N_2O_5), with part of CO is oxidized to CO_2 and SO_2 is oxidized to SO_3. The pentoxide forms nitric acid vapor as it contacts the water vapor in the gas, and similarly, sulfite acid, sulfate acid, bicarbonate, and carbonate acid vapors are formed. Then nitric acid vapor is absorbed in the scrubber as dilute nitric acid and is then neutralized by the dilute sodium carbonate in the scrubber forming sodium nitrate. Accordingly, sodium bicarbonate, sodium carbonate, sodium sulfite, sodium sulfate are produced. These dilute salts can be disposed to the sanitary sewer system for small-scale industrial boilers. For large-scale systems, byproduct recovery can be an option. In practice in order to obtain minimum NOx emissions, extra ozone is added to the system, which not only increases the system cost but also causes ozone slip. Therefore, a feedback system is installed at the top of the scrubber of the LTO process. When the NOx emissions are high, the feedback system can control the ozone

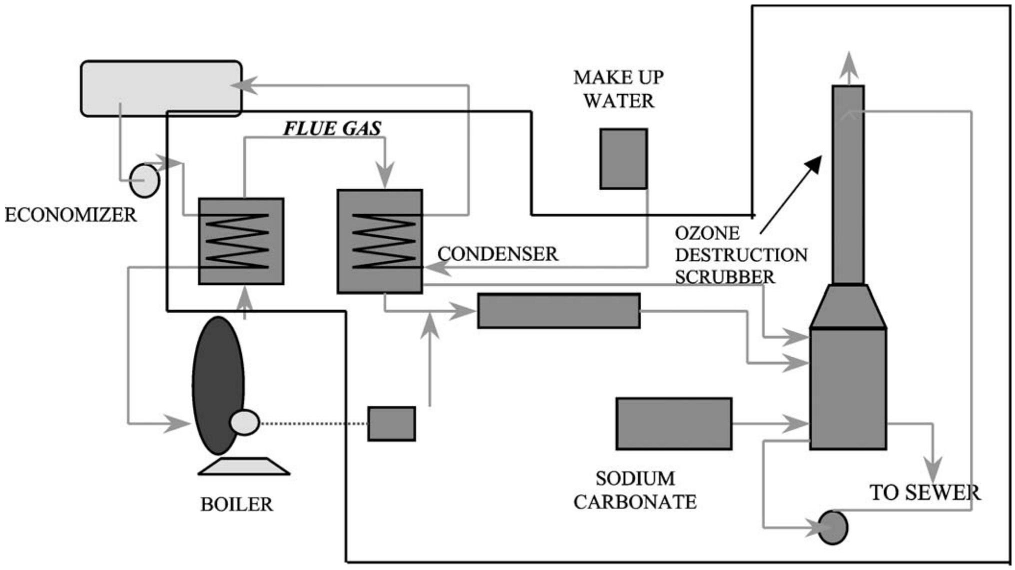


Figure 5. The low temperature oxidation (LTO) process diagram for small-scale industrial boiler applications.

generator to provide more ozone. When the ozone concentrations are high, the feedback system can inform the ozone generator to reduce ozone generation. In this way, ozone slip can be minimized. The drawbacks are (1) the sensor of the feedback system is quite expensive and requires high maintenance cost and (2) the system is actually operated dynamically instead of in steady state. Here our optimization approach to solve this problem is to add another constraint to make sure that the ozone slip is less than or equal to the allowable limit (which can be easily changed according to customer's request).

In our previous works, a detailed model of the LTO process based on a non-equilibrium modeling technique consisting of large-scale differential algebraic system was developed (Fu, 2000; Fu et al., 2000) and two major bottlenecks involved in modeling this system such as numerical difficulties and two-point boundary value problem were solved as well (Fu et al., 1999). The model is computationally intensive and MOP increases the computational load significantly. We had to use distributed computing system to obtain the optimal solutions in a reasonable amount of time. Therefore, this case study is a good candidate for testing the performance of the MINSOOP algorithm. It should be noted here that we are using the older LTO technology for this analysis. The new technology removes SOx and CO emissions more efficiently than the results showed in this paper. The multiobjective case study presented in this paper does not include the ozone destruction part of the scrubber in the detail model; the ozone slip is assumed to be less than 5 ppm in the current study. In the future, the model will include the ozone destruction reactions, and minimizing ozone slip would be included as an additional objective with ozone destruction scrubber parameters as additional decision variables.

4.1.1. Formulating the multiobjective optimization problem

In this case study, we would like to provide optimal design alternatives for the LTO process in order to reduce pollutant emissions of a 400 HP boiler operated at a 60% load condition (Suchak, 2000) with a 10713 pounds/hour of flue gas, with 39 ppm NOx, 500 ppm SOx, and 10 ppm CO at the temperature of 78°F and pressure of 1 atm. The capital cost for the system is spread over a 10-year period. The multiobjective optimization formulation for this case study is described in problem (6) below:

$$
\begin{aligned}
\text{minimize}\ & \text{NOx emissions},\\
& \text{SOx emissions},\\
& \text{Cost},\\
\text{subject to}\ & \text{Error in predicting concentrations of input liquid}\\
& \text{of the scrubber} \leqslant 10^{-6},\\
& \text{O}_3 \text{ emission} \leqslant 5 \text{ ppm},\\
& x_{Lj} \leqslant x_j \leqslant x_{Uj},\ j = 1, \ldots, 13.
\end{aligned}
\tag{6}
$$

Here the NOx and SOx emissions and cost are minimized concurrently. At the same time, the error in predicting concentrations of input liquid of the scrubber is less than or equal to 10^{-6} so as to ensure the model's accuracy and ozone emission is less than or equal to 5 ppm to guarantee low ozone slip. Here we choose 13 key decision variables that are identified to be important to the LTO process's performance, cost, and model's accuracy according to the engineering knowledge and previous modeling experience. The details of these decision variables are defined in table 4. It can be seen from table 4 that only 6 variables $(x_1, \ldots, x_6)$ out of the 13 decision variables are the real decision variables. The rest of them are intermediate variables used by the optimizer to ensure the model's accuracy due to the two-point value problem in the scrubber (Fu and Diwekar, 2003). The other variables are embedded in the model as this problem involves a large model with large number of stiff nonlinear differential equations. The ranges and the base design variables are determined by heuristics and experience with this process.

4.1.2. Generating the payoff table

The nonlinear optimizer is employed to solve three single objective optimization problems separately, keeping the error in predicting concentrations of input liquid of the scrubber less than or equal to 10^{-6} and ozone emissions less than or equal to 5 ppm for each of the three objectives. The optimizer is started from 50 different randomly generated initial points and the minimum value is retained as the optimal solution due to the nonlinear nonconvex nature of the problem. Then the values of the other two objectives are calculated at each of the three optimal solutions. These objective values are listed in the payoff table shown in table 5. In this way, an approximation of the potential ranges for all three objectives in the Pareto set is determined.

Table 4
The bounds and the base design values of the key decision variables for the multiobjective optimization case study of the LTO process.

Symbols x_j	Name of key decision variable x_j	Unit	Lower bound x_{Lj}	Upper bound x_{Uj}	Base design x_{oj}
x_1	Diameter of the oxidizer	m	0.5	1.5	0.762
x_2	Length of the oxidizer	m	2.0	20	24.4
x_3	Mole ratio of O_3/NOx	mole/mole	1.0	2.0	1.6
x_4	Diameter of the scrubber	m	0.5	1.5	0.914
x_5	Length of the scrubber	m	1.0	2.5	1.524
x_6	Liquid flow rate of the scrubber	gallon/min	50	150	85
x_7	Liquid output concentrations of $NaHCO_3$*	mole/mole of H_2O	1.0E−6	1.0E−2	**
x_8	Liquid output concentrations of Na_2CO_3*	mole/mole of H_2O	1.0E−6	1.0E−2	**
x_9	Liquid output concentrations of Na_2SO_4*	mole/mole of H_2O	1.0E−6	1.0E−2	**
x_{10}	Liquid output concentrations of Na_2SO_3*	mole/mole of H_2O	1.0E−6	1.0E−2	**
x_{11}	Liquid output concentrations of $NaNO_2$*	mole/mole of H_2O	1.0E−6	1.0E−2	**
x_{12}	Liquid output concentrations of $NaNO_3$*	mole/mole of H_2O	1.0E−6	1.0E−2	**
x_{13}	Liquid output concentrations of O_3*	mole/mole of H_2O	1.0E−6	1.0E−2	**

* Intermediate variables that are used by the optimizer to keep the error in predicting the input liquid concentrations in the scrubber (due to the two-point boundary value problem) less than or equal to 10^{-6} to ensure the model's accuracy.

** Values for these intermediate variables are automatically calculated by the optimizer with minimizing the error in predicting the input liquid concentrations in the scrubber as the objective.

Table 5
Payoff table for the case study of the LTO process.

Objective	Symbol i	NOx emissions (ppm)	SOx emissions (ppm)	Cost ($/hr)	P_{Error}*	Extra ozone (ppm)
Min NOx emission	1	4.907	129.7	16.65	1.0E−6	4.976
Min SOx emission	2	8.667	72.85	15.64	1.0E−6	5.0
Min cost	3	35.55	294.9	13.77	4.5E−7	2.227
Lower bound	Z_{iL}	4.907	72.85	13.77		
Upper bound	Z_{iU}	35.55	294.9	16.65		

* P_{Error} means error in predicting concentrations of input liquid of the scrubber.

4.1.3. Formulating a family of single objective optimization problems

Once the lower and upper bounds for each of the three objectives are obtained, the original multiobjective optimization problem is then transferred into a family of single objective optimization problems in the general form as follows.

$$\begin{array}{ll} \text{Minimize cost} & \\ \text{subject to} & \text{NOx emissions} \leqslant \mathit{Rh\_NOx}, \\ & \text{SOx emissions} \leqslant \mathit{Rh\_SOx}, \\ & \text{Error in predicting concentrations of input liquid} \\ & \text{of the scrubber} \leqslant 10^{-6}, \\ & \text{O}_3 \text{ emission} \leqslant 5 \text{ ppm}, \\ & x_{Lj} \leqslant x_j \leqslant x_{Uj},\ j = 1, \ldots, 13. \end{array} \tag{7}$$

In this case study, cost is selected as the objective to be minimized, NOx and SOx emissions are transferred into inequality constraints with two parametric right-hand sides, *Rh_NOx* and *Rh_SOx*, and these two new constraints are added to the original constraints of the multiobjective optimization problem. In this way, the original multiobjective optimization problem is transformed into a family of single objective optimization problems. Here we choose to generate 100 additional single objective optimization problems by efficiently and uniformly changing the bounds of *Rh_NOx* and *Rh_SOx* within their lower and upper bounds defined in table 5 (*Rh_NOx* within 4.907–35.55 ppm, *Rh_SOx* within 72.85–294.9 ppm) using the Hammersley sequence sampling technique. In the next subsection the nonlinear optimizer will be employed to find a set of minimum cost designs (optimal), which satisfy the corresponding limits for the NOx and SOx emissions specified in the current subsection.

4.1.4. Obtaining Pareto optimal solutions

Solving the appropriately formulated single objective optimization problems for every combination of the right-hand side values determined by the Hammersley sequence sampling technique using the nonlinear optimizer, a set of optimal design alternatives are obtained. This includes 103 (3 initial single objective optimization problems plus 100 additional transformed single objective optimization problems) optimal solutions. These optimal solutions form an approximation for the Pareto set of the LTO process for this case study. If a decision-maker knows his/her target, he/she can search from these 103 optimal designs to find the final design that is the most appropriate one to fit the decision-maker's implicit value function. However, in most of the cases, the decision-maker would rather know the characteristics of the problem and the tradeoffs information among objectives before making his/her choice, which will be provided in details in the next subsection.

In order to look into the details of the nonlinear part of the Pareto surface, the contour plot of the approximate Pareto surface is drawn in figure 6. Three major things catch our attention. First, designs within the small circle with a $15.1/hour cost, about 25 ppm NOx emissions, and 250 ppm SOx emissions, can be dominated by a large number of designs located down and to the left, which have the same or less cost, lower NOx and/or lower SOx emissions. Second, designs in the circle with a $14.4/hour cost at the top-right corner of the contour figure, can be conquered at least by designs in the same cost circle on its left, which have lower NOx emissions and similar SOx emissions, and also by designs with the same cost but lower SOx emissions, and similar NOx emissions

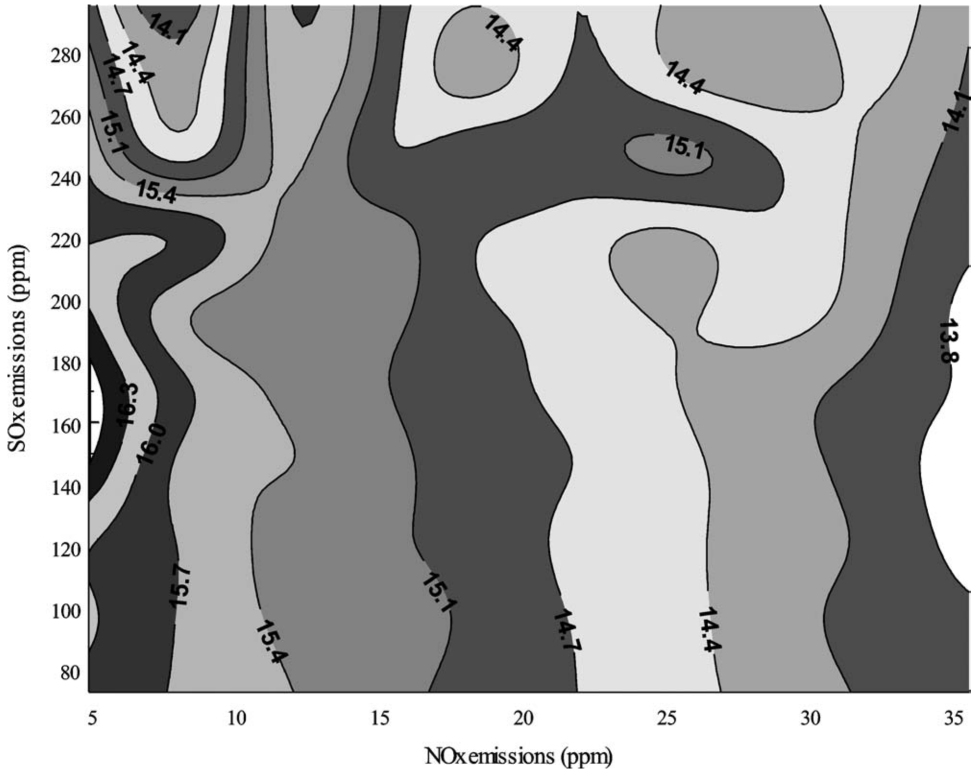


Figure 6. Contour plot of the approximate Pareto surface for the case study.

underneath. At this point, we can say that, by using multiobjective optimization approach we can take some advantage of the nonlinear non-convex feature of the problem to eliminate bad designs. Furthermore, figure 7 also shows that the cost tends to increase rapidly when the concentration of NOx emissions is low ($<$ 8 ppm) because the spacing of the \$0.3/hour cost contour lines, with respect to the NOx emissions, is much closer than in the higher NOx emissions region ($\geqslant$ 8 ppm). Another important issue that can be addressed by the multiobjective optimization approach is to provide explicit tradeoff information among different objectives and offer a reasonable number of attractive good designs for the decision-maker so that he/she can identify the final compromise design(s) for implementation. From the MOP decision and objective surfaces, we could identify four distinctive groups (I–IV) in these Pareto sets.

In the first group, all four designs are good candidates if the decision-maker wants to improve the base design in all objectives and satisfy the extra ozone constraint in the current case study. If the goal of the decision-maker is to see the potential ability of the LTO process for the SOx emissions removal and he/she is willing to sacrifice slightly in cost over the base design, then the two optimal designs in group II are better candidates. When the aim of the decision-maker is to reduce much more NOx emissions and have better greater SOx emissions removal than the base design with more tolerance for cost

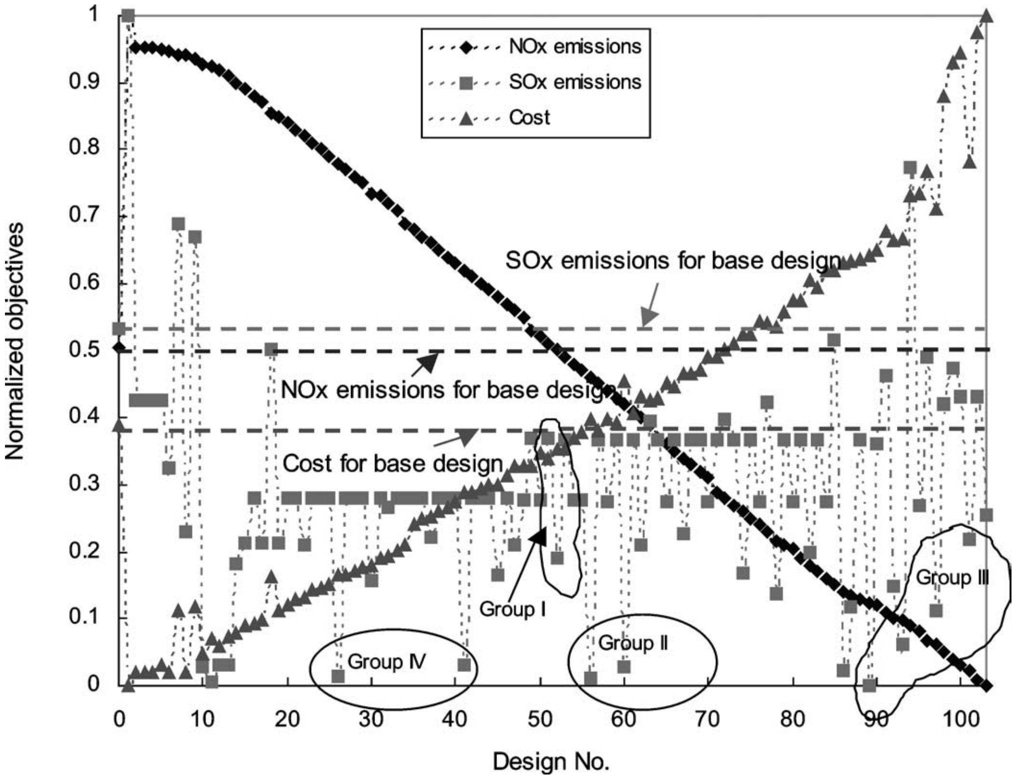


Figure 7. Normalized objectives of the 104 (the base design and 103 optimal designs) different designs for the case study of the LTO process.

increases, then the four optimal designs in group III are better choices. As the target of the decision-maker changes to reduce cost with a small sacrifice in NOx emissions, then the two optimal designs of group IV might sound attractive.

However, in-spite of the distinctive characteristics of the above four groups, there are also some common elements that can be found. They are summarized in that good designs tend to have (1) a larger diameter of the oxidizer, which can help to reduce more NOx emissions, (2) a smaller diameter of the scrubber which can cut the scrubber cost while reducing more SOx emissions, (3) a taller scrubber which can also remove more SOx emissions, (4) a larger liquid flow rate which can reduce both SOx and NOx emissions, and (5) a smaller ratio of scrubber cost/oxidizer cost, which indicates the tradeoff between the oxidizer and scrubber.

It is worth re-emphasizing that we tend to use the nonlinear, nonconvex nature of the problem to choose designs that have a much better SOx emissions reduction rate and have similar NOx emissions and cost compared to the designs nearby whenever this is possible (figure 7). Thus far, the original 103 optimal designs have been screened to a reasonable number of the potentially attractive designs in four different groups, which will help the decision-maker to identify his/her final choice.

5. Conclusions

This paper dealt with an important problem in multiobjective nonlinear programming, i.e., improving the computational efficiency of the MOP methods in order to solve large-scale real-world problems. Criteria used to compare different generating techniques are proposed, with the mean and variance of an approximate Pareto set calculated in order to evaluate the performance of a generating method for representing the whole Pareto set. A new nonlinear multiobjective optimization algorithm − MINSOOP, based on the Hammersley sequences sampling technique, has been developed and compared with the traditional constraint method intensively. Different numbers of sub-problems with different numbers of decision variables, constraints and objectives are solved for a set of nonlinear convex examples. Demonstration of the MINSOOP algorithm to obtain a representation of the whole Pareto set with nonlinear nonconvex multiobjective problems has also been given by a nonlinear nonconvex complete efficiency example and a real world case study.

The computational tests of the MINSOOP algorithm and the traditional constraint method to generate a true representation of the whole Pareto set have led us to make the following comments:

(1) The uniformity property of the Hammersley sequence sampling technique appears to be instrumental to the success of the MINSOOP algorithm.

(2) The convergence of the MINSOOP algorithm is much faster than with the traditional constraint method, which indicates significant computational savings by using MINSOOP algorithm.

(3) The performance of the new algorithm is not observed to be sensitive to the increased number of decision variables and constraints in terms of the number of nonlinear sub-problems to be solved for the same accuracy. However, increasing the number of decision variables and constraints will somehow increase the CPU time needed to solve each of the single objective nonlinear sub-problems. Therefore, the CPU time will actually increase with the increased number of variables and constraints.

(4) The computational saving of the MINSOOP algorithm versus the traditional constraint method increases dramatically when the number of objectives increases. This indicates that the MINSOOP algorithm is particularly fitting in situations where there are more than two objectives.

Experimental results obtained so far are encouraging. We have solved a number of relatively small problems having both nonlinear convex and nonconvex objectives. For all problems, the MINSOOP algorithm can converge to the "true" mean and variance of the Pareto set quite faster than the traditional constraint method. This illustrates that the new nonlinear multiobjective optimization algorithm can offer significant computational savings and better accuracy.

Acknowledgments and disclaimer

Although the research described in the article has been fully funded by the U.S. Environmental Protection Agency's STAR program through grant # R838128 and CR826085, it has not been subjected to any EPA review and therefore does not necessarily reflect the views of the Agency, and no official endorsement should be inferred.

Appendix A. The Hammersley points (Diwekar, 2003)

This appendix provides a definition of the Hammersley points and explicate a procedure for it's design. Any integer n can be written in radix-R notation (R is an integer) as follows:

$$n \equiv n_m n_{m-1} \ldots n_2 n_1 n_0 = n_0 + n_1 R + n_2 R^2 + \cdots + n_m R^m,$$

where $m = [\log_R n] = [\ln n / \ln R]$, the square brackets denote the integral part. A unique fraction between 0 and 1 called the *inverse radix number* can be constructed by reversing the order of the digits of n about the decimal point as follows:

$$\phi_R(n) = 0.n_0 n_1 n_2 \ldots n_m = n_0 R^{-1} + n_1 R^{-2} + \cdots + n_m R^{-m-1}.$$

The Hammersley points on a k-dimensional cube is given by the following sequence:

$$\vec{z}_k(n) = \left(\frac{n}{N}, \phi_{R_1}(n), \phi_{R_2}(n), \ldots, \phi_{R_{k-1}}(n)\right), \quad n = 1, 2, \ldots, N,$$

where $R_1, R_2, \ldots, R_{k-1}$ are the $k-1$ prime numbers where R_1 is the first prime number selected randomly. The Hammersley points are $\vec{x}_k(n) = 1 - \vec{z}_k(n)$.

References

Agrell, P.J., B.J. Lence, and A. Stam. (1998). "An Interactive Multicriteria Decision Model for Multipurpose Reservoir Management: The Shellmouth Reservoir." *Journal of Multi-Criteria Decision Analysis* 7(2), 61–86.

Armann, R. (1989). "Solving Multiobjective Programming Problems by Discrete Optimization." *Optimization* 20, 483–492.

Benson, H.P. and S. Sayin. (1997). "Towards Finding Global Representations of the Efficient Set in Multiple Objective Mathematical Programming." *Naval Research Logistics* 44(1), 47–67.

Chankong, V. and Y.Y. Haimes. (1983a). "Optimization-Based Methods for Multiobjective Decision-Making: An Overview." *Large Scale Systems* 5(1), 1–33.

Chankong, V. and Y.Y. Haimes. (1983b). *Multiobjective Decision Making Theory and Methodology*. New York: Elsevier Science.

Chankong, V., Y.Y. Haimes, J. Thadathil, and S. Zionts. (1985). "Multiple Criteria Optimization: A State of the Art Review." In Y.Y. Haimes and V. Chankong (eds.), *Decision Making with Multiple Objectives*, Lecture Notes in Economics and Mathematical Systems, Vol. 242, pp. 36–90. Berlin: Springer.

Cohon, J.L. (1978). *Multiobjective Programming and Planning*. New York: Academic Press.

Cohon, J., G. Scavone, and R. Solanki. (1988). "Multicriterion Optimization in Resources Planning." In W. Stadler (ed.), *Multicriteria Optimization in Engineering and in the Sciences*, pp. 117–160. New York: Plenum Press.

Cooper, C. (1998). "EPA Imposes 28% NOx Cut by 2007." *Chemical Engineering*, November, 54.

Diwekar, U. (2003). *Introduction to Applied Optimization*. Boston, MA: Kluwer Academic.

Diwekar, U. and E. Rubin. (1991). "Stochastic Modeling of Chemical Processes." *Computers and Chemical Engineering* 15, 105–119.

Eschenauer, H.A. (1988). "Multicriteria Optimization Techniques for Highly Accurate Focusing Systems." In W. Stadler (ed.), *Multicriteria Optimization in Engineering and in the Sciences*, pp. 309–354. New York: Plenum Press.

Evans, G.W. (1984). "An Overview of Techniques for Solving Multiobjective Mathematical Programs." *Management Science* 30(11), 1268–1282.

Ferreira, P.A.V. and M.E.S. Machado. (1996). "Solving Multiple-Objective Problems in the Objective Space." *Journal of Optimization Theory and Applications* 89(3), 659–680.

Fu, Y. (2000). "Process Design for the Environment: A Multiobjective Optimization Framework." Ph.D. Dissertation, Department of Civil and Environmental Engineering, Carnegie Mellon University, Pittsburgh, PA 15213.

Fu, Y. and U.M. Diwekar. (2003). "Cost Effective Environmental Management for Utilities." *Advances in Environmental Research* 8, 173–196.

Fu, Y., U.M. Diwekar, and N. Suchak. (2000). "Optimization Framework for Modeling the Low Temperature NOx Reduction Process." *Advances in Environmental Research* 3(4), 424–438.

Fu, Y., U.M. Diwekar, N. Suchak, and W. Rohrer. (1999). "Modeling the Best Available Control Technology: Numerical Difficulties and the Optimization Framework." In *Proceedings of the AIChE Annual Meeting*, Dallas, TX.

Fu, Y., U.M. Diwekar, D. Young, and H. Cabezas. (2000). "Process Design for Environment: A Multiobjective Framework under Uncertainty." *Journal of Clean Products and Processes* 2, 92–105.

Haimes, Y.Y., L.S. Lasdon, and D.A. Wismer. (1971). "On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization." *IEEE Transactions on Systems, Man, and Cybernetics* 1, 296–297.

Hammersley, J.M. (1960). "Monte Carlo Methods for Solving Multivariate Problems." *Annals of New York Academy of Science* 86, 844–874.

Hwang, C.L. and A.S.M. Masud. (1979). *Multiple Objective Decision Making – Methods and Applications: A State-of-the-Art Survey*, Lecture Notes in Economics and Mathematical Systems, Vol. 164. Berlin: Springer.

Johnson, T.L. and U.M. Diwekar. (2001). "Hanford Waste Blending and the Value of Research: Stochastic Optimization as a Policy Tool." *Journal of Multi-Criteria Decision Analysis* 10, 87–98.

Kalagnanam, J.R. and U.M. Diwekar. (1997). "An Efficient Sampling Technique for Off-Line Quality Control." *Technometrics* 39(3), 308–319.

Kim, K. and U.M. Diwekar. (2002). "Hammersley Stochastic Annealing: Efficiency Improvement for Combinatorial Optimization under Uncertainty." *IEE Transactions on Operations Research* 34(9), 761–778.

Kim, K. and U.M. Diwekar. (2003). "Fractal Dimension Approach to Error Characterization in Sampling." In *INFORMS Annual Meeting*, Atlanta, GA.

Kumar, P., N. Singh, and N.K. Tewari. (1991). "A Nonlinear Goal Programming Model for Multistage, Multiobjective Decision Problems with Application to Grouping and Loading Problem in a Flexible Manufacturing System." *European Journal of Operational Research* 53(2), 166–171.

Lampinen, J. (2000). "Multiobjective Nonlinear Pareto Optimization." A Pre-Investigation Report, Lappeenranta University of Technology, Laboratory of Information Processing, Lapperanta, Finland.

Miettinen, K.M. (1999). *Nonlinear Multiobjective Optimization*. Norwell, MA: Kluwer Academic.

NWTRB. (1996). "Disposal and Storage of Spent Nuclear Fuel: Finding the Right Balance." Arlington, VA: Nuclear Waste Technical Review Board.

Ohkubo, S., P.B.R. Dissanayake, and K. Taniwaki. (1998). "An Approach to Multicriteria Fuzzy Optimization of a Prestressed Concrete Bridge System Considering Cost and Aesthetic Feeling." *Structural Optimization* 15(2), 132–140.

Olson, D.L. (1993). "Tchebycheff Norms in Multiobjective Linear Programming." *Mathematical and Computer Modeling* 17(1), 113–124.

Osyczka, A. (1984). *Multicriterion Optimization in Engineering with FORTRAN Programs*. Chichester: Ellis Horwood.

Rosenthal, R.E. (1985). "Principles of Multiobjective Optimization." *Decision Sciences* 16(2), 133–152.

Sahin, K. and U.M. Diwekar. (2002). "Quasi-Monte Carlo Method for Faster Molecular Simulations." In *Proceedings of AIChE Annual Meeting*, Indianapolis, IN.

Sayin, S. (2000). "Measuring the Quality of Discrete Representation of Efficient Sets in Multiple Objective Mathematical Programming." *Mathematical Programming* 87, 543–560.

Sayin, S. (2003). "A Procedure to Find Discrete Representations of the Efficient Set with Specified Convergence Errors." *Operations Research* 51(3), 427–436.

Schy, A.A. and D.P. Giesy. (1988). "Multicriteria Optimization Methods for Design of Aircraft Control Systems." In W. Stadler (ed.), *Multicriteria Optimization in Engineering and in the Sciences*, pp. 225–262. New York: Plenum Press.

Silverman, J., R.E. Steuer, and A.W. Whisman. (1988). "A Multi-Period, Multiple Criteria Optimization System for Manpower Planning." *European Journal of Operational Research* 34(2), 160–170.

Sobol, I.M. (1992). "A Global Search for Multicriterial Problems." In A. Goicoechea, L. Duckstein, and S. Zionts (eds.), *Proceedings of the 9th International Conference, Multiple Criteria Decision Making*, pp. 401–412. New York: Springer.

Stadler, W. (ed.). (1988). *Multicriteria Optimization in Engineering and in the Sciences*. New York: Plenum Press.

Starkey, J.M., S. Gray and D. Watts. (1988). "Vehicle Performance Simulation and Optimization Including Tire Slid." *ASME 881733.*

Steuer, R.E. (1986). *Multiple Criteria Optimization: Theory, Computation, and Applications*. New York: Wiley.

Stewart, T.J. (1992). "A Critical Survey on the Status of Multiple Criteria Decision Making Theory and Practice." *Omega* 20(5–6), 569–586.

Suchak, N.J. (May 2000). Personal communication.

Tamiz, M. and D.D. Jones. (1996). "An Overview of Current Solution Methods and Modeling Practices in Goal Programming." In M. Tamiz (ed.), *Multiobjective Programming and Goal Programming: Theories and Applications*. Lecture Notes in Economics and Mathematical Systems, Vol. 432, pp. 198–211. Berlin: Springer.

Wood, E., N.P. Greis, and R.E. Steuer. (1982). "Linear and Nonlinear Applications of the Tchebycheff Metric to the Multi-Criteria Water Allocation Problem." In S. Rinaldi (ed.), *Environmental Systems Analysis and Management*, pp. 363–376. Amsterdam: North-Holland.

Yu, P.L. (1985). *Multiple-Criteria Decision Making Concepts, Techniques, and Extensions*. New York: Plenum Press.

Zeleny, M. (1974). *Linear Multiobjective Programming.* Lecture Notes in Economics and Mathematical Systems, Vol. 95. Berlin: Springer.

Zeleny, M. (1982). *Multiple Criteria Decision Making*. New York: McGraw-Hill.

Zoints, S. (1989). "Multiple Criteria Mathematical Programming: An Updated Overview and Several Approaches." In B. Karpak and S. Zionts (eds.), *Multiple Criteria Decision Making and Risk Analysis Using Microcomputers*, pp. 7–60. Berlin: Springer.