IIE Transactions
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/uiie20

Intelligent interactive multiobjective optimization method and its application to reliability optimization

Hong-Zhong Huang [a], Zhigang Tian [b] & Ming J. Zuo [b]
[a] Mechatronics Engineering, University of Electronic Science and Technology of China, Chengdu, 610054, China E-mail:
[b] Department of Mechanical Engineering, University of Alberta, Edmonton, AB, T6G 2G8, Canada E-mail:

Version of record first published: 23 Feb 2007.

IIE Transactions (2005) **37**, 983–993

ISSN: 0740-817X print / 1545-8830 online
DOI: 10.1080/07408170500232040

Intelligent interactive multiobjective optimization method and its application to reliability optimization

HONG-ZHONG HUANG[1], ZHIGANG TIAN[2] and MING J. ZUO[2]

[1] *Mechatronics Engineering, University of Electronic Science and Technology of China, Chengdu, 610054, China*
E-mail: hzhuang@uestc.edu.cn
[2] *Department of Mechanical Engineering, University of Alberta, Edmonton, AB, T6G 2G8, Canada*
E-mail: ztian@ualberta.ca or ming.zuo@ualberta.ca

Received May 2003 and accepted September 2004

In most practical situations involving reliability optimization, there are several mutually conflicting goals such as maximizing the system reliability and minimizing the cost, weight and volume. This paper develops an effective multiobjective optimization method, the Intelligent Interactive Multiobjective Optimization Method (IIMOM). In IIMOM, the general concept of the model parameter vector is proposed. From a practical point of view, a designer's preference structure model is built using Artificial Neural Networks (ANNs) with the model parameter vector as the input and the preference information articulated by the designer over representative samples from the Pareto frontier as the desired output. Then with the ANN model of the designer's preference structure as the objective, an optimization problem is solved to search for improved solutions and guide the interactive optimization process intelligently. IIMOM is applied to the reliability optimization problem of a multi-stage mixed system with five different value functions simulating the designer in the solution evaluation process. The results illustrate that IIMOM is effective in capturing different kinds of preference structures of the designer, and it provides a complete and effective solution for medium- and small-scale multiobjective optimization problems.

1. Introduction

In most practical situations involving reliability optimization, there are several mutually conflicting goals such as maximizing the system reliability and minimizing the cost, weight and volume. Sakawa (1978) considered a multiobjective formulation to maximize the reliability and minimize the cost for reliability allocation by using a surrogate worth trade-off method. Inagaki *et al.* (1978) solved a different problem to maximize the reliability and minimize the cost and weight by using an interactive optimization method. The multiobjective reliability apportionment problem for a two-component series system has been analyzed by Park (1987) using fuzzy logic theory. Dhingra (1992) and Rao and Dhingra (1992) researched the reliability and redundancy apportionment problem for a four-stage and a five-stage overspeed protection system, using crisp and fuzzy multiobjective optimization approaches respectively. Ravi *et al.* (2000) modeled and analyzed the problem of optimizing the reliability of complex systems as a fuzzy multiobjective optimization problem.

It is very difficult for a designer to specify accurately his/her preference on the goals *a priori* in multiobjective reliability optimization problems. The most effective methods have been interactive procedures (Gardiner and Steuer, 1994), which typically include alternate solution generation and solution evaluation phases. There are three key issues in the interactive multiobjective optimization method (Sun *et al.*, 1996): (i) how to elicit preference information from the designer over a set of candidate solutions; (ii) how to represent the designer's preference structure in a systematic manner; and (iii) how to use the designer's preference structure to guide the search for improved solutions.

Current interactive multiobjective optimization methods include STEM, the Geoffrion-Dyer-Feinberg procedure, the visual interactive approach, the Tchebycheff method, the Zionts-Wallenius method, the reference point method (Gardiner and Steuer, 1994) and the interactive FFANN procedure (Sun *et al.*, 1996, 2000). Most of the methods above do not made full use of the designer's preference information on the generated solutions, and therefore cannot build the model of the designer's preference structure systematically. In Sun *et al.* (1996), an Artificial Neural Network (ANN) model of the designer's preference structure is built with the objective function value vector acting as the input and the corresponding preference value acting as the desired output. An optimization problem is solved with the ANN model with the objective of searching for improved solutions. Nevertheless, the improved solutions found in this way cannot be shown to be Pareto solutions, and neither can the final solution. In Sun *et al.* (2000), an ANN model of the designer's preference structure is built in the same way. This is then used to evaluate the Pareto solutions generated with Augment weighted Tchebycheff programs (AWTPs) in order to pick the 50% of the solutions with the highest preference values and present them to the designer for evaluation. The use of the ANN model reduces the burden on the designer of evaluating the generated solutions, but it could not help search for improved solutions, which is vital in interactive multiobjective optimization procedures.

This paper develops an effective multiobjective optimization method, the Intelligent Interactive Multiobjective Optimization Method (IIMOM), which is characterized by the way in which the designer's preference structure model is built and used to guide the search for improved solutions. In IIMOM, the general concept of the model parameter vector, which refers to the parameter vector determined by the designer in the multiobjective optimization model (such as the weight vector in the weighted-sum method), is proposed. From a practical point of view, the designer's preference structure model is built using an ANN with the model parameter vector acting as the input and the preference information articulated by a designer over representative samples from the Pareto frontier as the desired output. Then with the ANN model of the designer's preference structure as the objective, an optimization problem is solved to search for improved solutions. Two key advantages of IIMOM are: (i) the ANN model of the designer's preference structure can guide the designer towards exploring the most interesting parts of the Pareto frontier efficiently and accurately; and (ii) the improved solutions generated at each iteration are Pareto solutions, which is in stark contrast to the method presented in Sun *et al.* (1996).

IIMOM is applied to the reliability optimization problem of a multi-stage mixed system. Five different value functions are used to simulate the designer in the solution evaluation process. The results illustrate that IIMOM is effective in capturing different kinds of preference structures of the designer, and it is an effective tool for the designer to find the most satisfying solution.

2. Multiobjective optimization problem

2.1. *Problem formulation*

A general multiobjective optimization problem consists in finding the design variables that optimize m different objectives over the feasible design space. A mathematical formulation of the multiobjective optimization problem is:

$$\begin{aligned} &\min \mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})\}, \\ &\text{subject to } \mathbf{x} \in \mathrm{X}, \end{aligned} \tag{1}$$

where $\mathbf{x}$ is an n-dimensional vector of design variables, X is the feasible design space, $f_i(\mathbf{x})$ is the objective function of the ith design objective and $\mathbf{f(x)}$ is the design objective vector.

2.2. *Pareto solution*

A design variable vector $\mathbf{x}^{\mathrm{P}}$ is said to be a Pareto solution if there exists no feasible design variable vector $\mathbf{x}$ that would decrease some objective functions without causing a simultaneous increase in at least one other objective function. Mathematically, a solution $\mathbf{x}^{\mathrm{P}}$ is said to be a Pareto solution if for any $\mathbf{x} \in \mathrm{X}$ satisfying $\mathbf{f}_j(\mathbf{x}) < \mathbf{f}_j(\mathbf{x}^{\mathrm{P}})$, $\mathbf{f}_k(\mathbf{x}) > \mathbf{f}_k(\mathbf{x}^{\mathrm{P}})$, for at least one other objective $k \neq j$.

The set of all Pareto solutions of a multiobjective optimization problem is known as the Pareto frontier (or Pareto set), which is denoted by N. It is evident that the final solution of a multiobjective optimization problem should be selected from the Pareto frontier.

2.3. *Weighted-sum method*

The weighted-sum method is one of the most widely used solution methods for multiobjective optimization problems. It converts a multiobjective optimization problem into a single-objective optimization problem using a weighted sum of all the objective functions as the single objective. The mathematical model of the weighted-sum method takes the form of:

$$\begin{aligned} &\min\ f = \sum_{i=1}^{m} w_i f_i(\mathbf{x}), \\ &\text{subject to } \mathbf{x} \in \mathrm{X}, \end{aligned} \tag{2}$$

where w_i is the weight of objective i, and

$$\sum_{i=1}^{m} w_i = 1,\ w_i \geq 0,\ i = 1, 2, \ldots, m.$$

2.4. *AWTPs*

The AWTP is another widely used solution method for multiobjective optimization problems. The mathematical model of an AWTP takes the form:

$$\begin{aligned} &\min\ \alpha + \rho \sum_{i=1}^{m} (1 - \mathrm{z}_i), \\ &\text{subject to}\ \ \alpha \geq \lambda_i (1 - \mathrm{z}_i), \forall i, \\ &\mathrm{z}_i = \frac{f_i(x) - f_i^{\mathrm{nadir}}}{f_i^{\mathrm{ideal}} - f_i^{\mathrm{nadir}}}, \forall i, \\ &x \in \mathrm{X} \end{aligned} \tag{3}$$

where ρ is a small positive scalar; f^{ideal} is the utopian point, that is, f_i^{ideal} is the optimization result with the ith design objective as the objective function and as $x \in \mathrm{X}$ constraints; λ_i is the weight of the design objective i, and satisfies $\sum_{i=1}^{m}, \lambda_i = 1$, and $\lambda_i \geq 0$, $i = 1, 2, \ldots, m$; f_i^{nadir} is the worst value of the ith objective function (the worst value is the maximum value since we desire to minimize this objective function) among all the points in the Pareto frontier. For multiobjective linear programming problems, f_i^{nadir} can be evaluated with the method provided by Korhonen *et al.* (1997). For nonlinear programming problems, f_i^{nadir} can be estimated by the optimization result with the minus of the ith design objective as the objective function and $x \in X$ as constraints, or it can be estimated by simply being assigned a value based on experience.

Some interactive multiobjective optimization methods, such as the Tchebycheff method, WIERZ and SATIS, are based on AWTPs (Gardiner and Steuer, 1994). By changing the weight vector $\boldsymbol{\lambda}$ in AWTPs, each point of N can be reached (Chen *et al.*, 1997). Therefore, an AWTP is effective tool to construct interactive multiobjective optimization methods.

3. Designer's preference structure model

3.1. *Model parameter vector*

The general concept of the model parameter vector is proposed in this section. The model parameter vector refers to the parameter vector determined by the designer in the multiobjective optimization model, such as the weight vector $[w_1, w_2, \ldots, w_m]$ in the weighted-sum method and the weight vector $[\lambda_1, \lambda_2, \ldots, \lambda_m]$ in AWTP-based methods.

For a specific multiobjective optimization problem, the design variables, objective functions and constraints will have already been determined through analysis and modeling. The designer can therefore only manipulate the model parameter vector. That is, in a specific multiobjective optimization problem, once the model parameter vector is determined, the final solution of the problem will also be determined.

3.2. *General multiobjective optimization procedure*

The general multiobjective optimization procedure is shown in Fig. 1.

First, the problem is analyzed and the design variables, objective functions and the constraints are determined. Then the model parameter vector is set by the designer. Optimization is conducted and the solution is obtained. If the designer is satisfied with the obtained solution, the optimization procedure is terminated. Otherwise, the designer will modify the model parameter vector and conduct the next iteration of optimization.

There is one problem in the general multiobjective optimization procedure. The designer can control the model parameter vector, but he cannot control the generated objective function vector of the optimization solution with respect to the model parameter vector. The objective function vector of the optimization solution determines the designer's preference on the solution. Therefore, the general

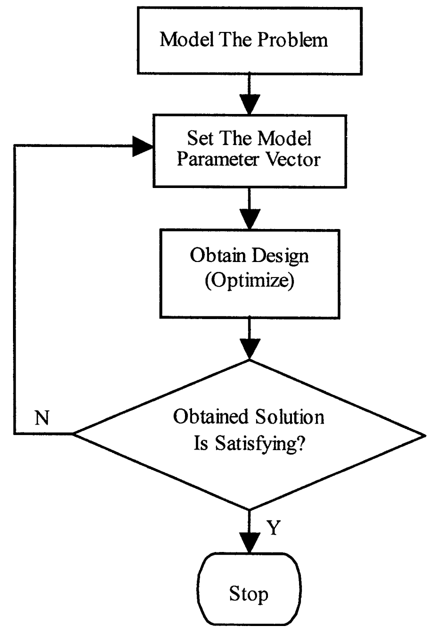

Fig. 1. The general multiobjective optimization procedure.

multiobjective optimization procedure typically requires many iterations on the choice of the model parameter vector and often provides no clear guidance as to how to converge to the right model parameter vector. For instance, consider the case where the weighted-sum method is used in the multiobjective optimization procedure. If the designer is not satisfied with the obtained solution and desires to improve objective i, the weight w_i should be increased relatively. However, how much w_i should be relatively increased is unknown. The convergence of the optimization procedure cannot be guaranteed. Therefore, an interactive multiobjective optimization method is needed which is able to intelligently guide the designer to explore the most relevant part of the Pareto frontier efficiently and accurately and to converge to a satisfactory solution.

Another important point implied in Fig. 1 is the vital role that the model parameter vector plays in the general multiobjective optimization procedure. The model parameter vector is the only item the designer can manipulate. The designer modifies the model parameter vector to express his preference on the generated solution and to generate new solutions.

3.3. *Designer's preference structure model*

The designer's preference structure represents the designer's preferences on the design objectives and their trade-off. After a solution is generated, the designer could express his preference information (e.g., assigned preference value) on the solution based on its objective functions' values. The process of how the preference information is elicited is depicted in Fig. 2.

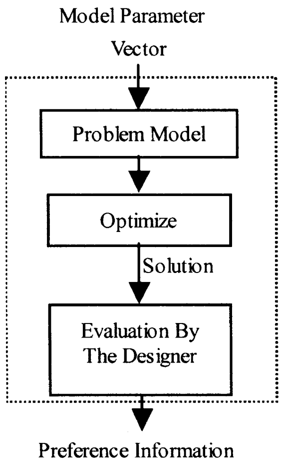

Fig. 2. The preference information elicitation process.

A direct approach to model the designer's preference structure is to build a model mapping the objective function value vector to the preference information, that is, to simulate the function block of "Evaluation By The Designer" in Fig. 2. ANNs have been used to build the designer's preference structure model in this way by Lu *et al.* (1955), Sun *et al.* (1996, 2000) and Stam *et al.* (1996). The method is easy to understand, but a designer's preference structure model built in this way is difficult to use to help guide the designer effectively in the subsequent optimization iterations, because the input of the model (the generated solution) cannot be controlled directly. For example, in the method proposed by Sun *et al.* (1996), an optimization problem is solved using an ANN model of the designer's preference structure as the objective to search for improved solutions. Nevertheless, the improved solutions found in this way cannot be shown to be Pareto solutions, and neither can the final solution.

For the preference information elicitation process in Fig. 2, it can be seen that the model parameter vector is the input and the preference information is the output. And the designer can modify the model parameter vector in order to achieve the solution that best satisfies his preference. Therefore, from a practical point of view, it is proposed in this paper that the designer's preference structure model should be built using an ANN that maps the model parameter vector to the preference information, that is, it simulates the function blocks in the broken line frame in Fig. 2.

ANNs have shown their ability to represent complex nonlinear mapping using a set of available data, so this technique is used to model the designer's preference structure, that is, to represent the mapping from the model parameter vector to the preference information. The ANN model of the designer's preference structure is trained with the Pareto solutions generated during the interactive optimization procedure and their corresponding preference information acts as the training set.

The model parameter vector is the vital factor in interactive optimization and also is the vital factor in building the designer's preference structure model, but it has always been overlooked in previous research. The ANN model of the designer's preference structure in this paper is easy to use because the input of the model (the model parameter vector) can be directly controlled. Also there is no evidence that an ANN model built in this way is more complex than the ANN model mapping the objective function value vector to the preference information, because both of them use a set of generated data to approximate nonlinear relationships using ANNs. The ANN model of the designer's preference structure plays a vital role in the IIMOM procedure presented in the following section.

3.4. *Preference information elicitation*

The preference information that acts as the output of the ANN model of the designer's preference structure, can be elicited in two ways (Sun *et al.*, 1996). The designer determines a Pareto solution's preference information by either directly assigning a preference "value" or by making pairwise comparisons among the generated Pareto solutions. The elicited preference information can be represented by a numerical value, the so-called preference value.

4. The IIMOM procedure

The IIMOM developed in this paper is based on the AWTPs formulated in Equation (3). The weight vector $[\lambda_1, \lambda_2, \ldots, \lambda_m]$ is the model parameter vector in the AWTPs. The IIMOM procedure is shown in Fig. 3 and is specified step-by-step below, followed by comments about its different steps.

Step 1. $l_i^{(h)}$ and $u_i^{(h)}$ denote respectively the lower and upper boundaries of weight λ_i at iteration h. $[l_i^{(h)}, u_i^{(h)}] \subseteq [0, 1], \forall i, \forall h$. Let $[l_i^{(1)}, u_i^{(1)}] = [0, 1], \forall i$, and a more specific $[l_i^{(1)}, u_i^{(1)}] \subset [0, 1], \forall i$ will be helpful for efficient convergence of the IIMOM procedure. $\boldsymbol{\Lambda}^{(h)}$ denotes the weight vector space at iteration h:

$$\boldsymbol{\Lambda}^{(h)} = \left\{ \boldsymbol{\lambda} \left| \sum_{i=1}^{m} \lambda_i = 1, \lambda_i \in \left[l_i^{(h)}, u_i^{(h)} \right] \forall i \right. \right\}$$

Specify the weight vector space reduction factor r, the number of Pareto solutions K to be evaluated at each iteration, and the structure of the ANN model of the designer's preference structure. Calculate f^{ideal}, and determine f_i^{nadir}.

Step 2. In the weight vector space, K weight vectors are randomly generated. If $h > 1$, the first weight vector is replaced by the best weight vector $\boldsymbol{\lambda}^{(\text{Opt},h-1)}$ obtained at the previous iteration. The best weight vector $\boldsymbol{\lambda}^{(\text{Opt},h-1)}$ is obtained by solving an optimization problem with the ANN model

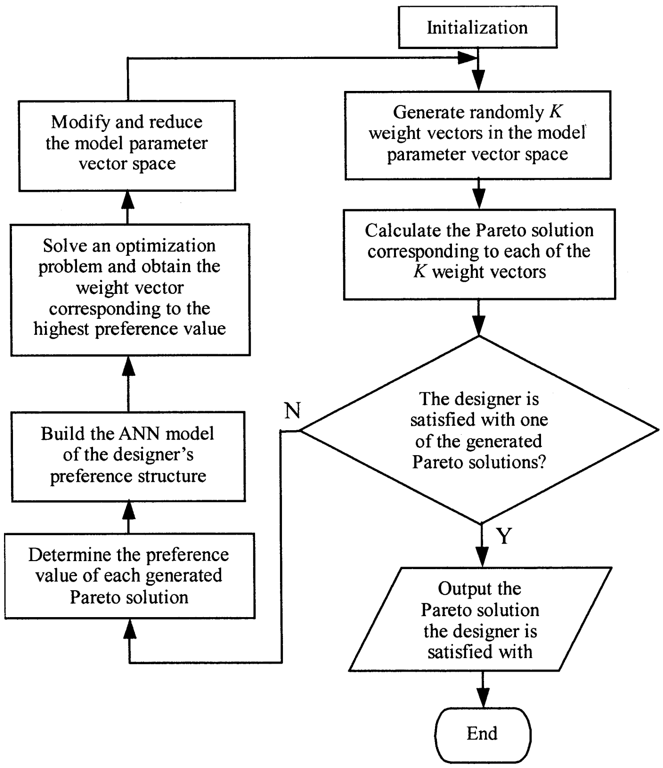

Fig. 3. The flowchart of the IIMOM procedure.

of the designer's preference structure as the objective. It refers to the weight vector with respect to the highest preference value in the ANN model at iteration h.

Step 3. Solve one AWTP problem for each of the K weight vectors to obtain K Pareto solutions.

Step 4. If the designer is satisfied with one of the Pareto solutions, that Pareto solution is output and the interactive optimization procedure is terminated. Otherwise, go to Step 5.

Step 5. The Pareto solutions obtained at the current iteration are presented to the designer, and their preference values are evaluated.

Step 6. With the Pareto solutions obtained in the last several iterations as the training set, the weight vector as the input, and the corresponding preference value as the desired output, a feed-forward neural network is trained to obtain the ANN model of the designer's preference structure.

Step 7. With the ANN model of the designer's preference structure as the objective function, the optimization problem shown in Equation (4) is solved to obtain the best weight vector $\boldsymbol{\lambda}^{(\mathrm{Opt},h)}$ with respect to the highest preference value:

$$\begin{aligned} &\max \quad \mathrm{ANN}(\boldsymbol{\lambda}), \\ &\text{subject to} \quad \boldsymbol{\lambda} \in \boldsymbol{\Lambda}^{(h)}, \end{aligned} \tag{4}$$

where $\mathrm{ANN}(\boldsymbol{\lambda})$ represents the preference value calculated using the ANN model of the designer's preference structure when the input weight vector is $\boldsymbol{\lambda}$.

Step 8. Let $P^{(\mathrm{Best},h)}$ denote the Pareto solution with respect to the highest preference value among all the Pareto solutions that have been generated until the current iteration, and $P^{(\mathrm{Opt},h)}$ denote the Pareto solution obtained in Step 7. $\boldsymbol{\lambda}^{(\mathrm{Best},h)}$ and $\boldsymbol{\lambda}^{(\mathrm{Opt},h)}$ are the weight vectors with respect to $P^{(\mathrm{Best},h)}$ and $P^{(\mathrm{Opt},h)}$ respectively. Let:

$$\boldsymbol{\lambda}^{(\mathrm{Center},h)} = \frac{\boldsymbol{\lambda}^{(\mathrm{Best},h)} + \boldsymbol{\lambda}^{(\mathrm{Opt},h)}}{2}. \tag{5}$$

Modify $l_i^{(h+1)}$ and $u_i^{(h+1)}$ for each design objective i to determine the new weight vector space $\boldsymbol{\Lambda}^{(h+1)}$:

If

$$\left|\lambda_i^{(\mathrm{Opt},h)} - \lambda_i^{(\mathrm{Best},h)}\right| > r^h$$

then

$$\left[l_i^{(h+1)}, u_i^{(h+1)}\right] = \left[\min\left(\lambda_i^{(\mathrm{Opt}),h}, \lambda_i^{(\mathrm{Best},h)}\right), \max\left(\lambda_i^{(\mathrm{Opt}),h}, \lambda_i^{(\mathrm{Best,h})}\right)\right],$$

if

$$\left|\lambda_i^{(\mathrm{Opt},h)} - \lambda_i^{(\mathrm{Best},h)}\right| \le r^h \quad \text{and} \quad \left(\lambda_i^{(\mathrm{Center},h)} - \frac{r^h}{2}\right) \le 0$$

then

$$\left[l_i^{(h+1)}, u_i^{(h+1)}\right] = \left[0, \lambda_i^{(\mathrm{Center},h)} + \frac{r^h}{2}\right]$$

if

$$\left|\lambda_i^{(\mathrm{Opt},h)} - \lambda_i^{(\mathrm{Best},h)}\right| \le r^h \quad \text{and} \quad \left(\lambda_i^{(\mathrm{Center},h} + \frac{r^h}{2}\right) \ge 1$$

then

$$\left[l_i^{(h+1)}, u_i^{(h+1)}\right] = \left[\lambda_i^{(\mathrm{Center},h)} - \frac{r^h}{2}, 1\right]$$

otherwise

$$\left[l_i^{(h+1)}, u_i^{(h+1)}\right] = \left[\lambda_i^{(\mathrm{Center},h)} - \frac{r^h}{2}, \lambda_i^{(\mathrm{Center},h)} + \frac{r^h}{2}\right]. \tag{6}$$

Then go to Step 2 and conduct the next iteration of IIMOM.

IIMOM has two key advantages:

1. the ANN model of the designer's preference structure can guide the designer to explore the part of the Pareto frontier of interest to him efficiently and accurately. IIMOM has an outstanding convergence performance. During the IIMOM procedure, the ANN model of the designer's preference structure will become more and

more accurate around the part of the Pareto frontier of interest to the designer.
2. The improved solutions generated at each iteration are Pareto solutions, which is in stark contrast to the method presented by Sun *et al.* (1996).

5. Reliability optimization problem

In this section, IIMOM is applied to the reliability optimization problem of a multi-stage mixed system. Five different value functions are used to simulate the designer in the solution evaluation process in order to illustrate the effectiveness of IIMOM in capturing different kinds of preference structures of the designer and finding the most satisfying solution.

5.1. *Problem definition*

The multiobjective reliability optimization problem is taken from Sakawa (1982) and Ravi *et al.* (2000) who model the problem as a fuzzy multiobjective optimization problem. A multi-stage mixed system is considered, where the problem is to allocate the optimal reliabilities $r_i, i = 1, 2, 3, 4$ of four components whose redundancies are specified. The multiobjective optimization model of the problem takes the following form (Ravi *et al.*, 2000):

$$\max R_S, \min C_S, \min W_S,$$

subject to

$$V_S = \sum_{j=1}^{4} V_j n_j \leq 65, \quad P_S \leq 12\,000, \tag{7}$$

where R_S, C_S, W_S, V_S are the reliability, cost, weight and volume of the system.

$$\begin{aligned} P_S &= W_S \times V_S, \\ R_S &= \prod_{j=1}^{4} [1 - (1 - r_j)^{n_j}], \\ C_S &= \sum_{j=1}^{4} C_j n_j, \\ W_S &= \sum_{j=1}^{4} W_j n_j, \end{aligned} \tag{8}$$

$$\begin{aligned} C_j &= \alpha_j^{c} \left[\log_{10} \left(\frac{\beta_j^{c}}{1 - r_j} \right) \right]^{\gamma_j^{c}}, \\ W_j &= \alpha_j^{w} \left[\log_{10} \left(\frac{\beta_j^{w}}{1 - r_j} \right) \right]^{\gamma_j^{w}}, \\ V_j &= \alpha_j^{w} \left[\log_{10} \left(\frac{\beta_j^{v}}{1 - r_j} \right) \right]^{\gamma_j^{v}}, \\ \alpha_j^{c} &= 8.0, \alpha_j^{w} = 6.0, \alpha_j^{v} = 2.0, \\ \gamma_j^{c} &= 2.0, \gamma_j^{w} = 0.5, \gamma_j^{v} = 0.5, \end{aligned} \tag{9}$$

$$\begin{aligned} &\beta_1^{c} = 2.0, \beta_2^{c} = 10.0, \beta_3^{c} = 3.0, \beta_4^{c} = 18.0, \\ &\beta_1^{w} = 3.0, \beta_2^{w} = 2.0, \beta_3^{w} = 10.0, \beta_4^{w} = 8.0, \\ &\beta_1^{v} = 2.0, \beta_2^{v} = 2.0, \beta_3^{v} = 6.0, \beta_4^{c} = 8.0, \\ &n_1 = 7, n_2 = 8, n_3 = 7, n_4 = 8, \end{aligned} \tag{10}$$

where all the values use the corresponding SI units.

IIMOM is applied to the formulated multiobjective reliability optimization problem. Let $\lambda_i^{(1)} \in [0, 1], i = 1, 2, \lambda_3^{(h)} = 1 - \lambda_1^{(h)} - \lambda_2^{(h)}$. The ideal and nadir value of the three objectives are determined as follows:

$$\begin{aligned} R_S^{\text{ideal}} &= 1, \quad R_S^{\text{nadir}} = 0.9, \\ C_S^{\text{ideal}} &= 0, \quad C_S^{\text{nadir}} = 550, \\ W_S^{\text{ideal}} &= 0, \quad W_S^{\text{nadir}} = 350. \end{aligned} \tag{11}$$

In the framework of the AWTPs formulated in Equation (3), we have that:

$$\begin{aligned} z_1 &= \frac{R_S - R_S^{\text{nadir}}}{R_S^{\text{ideal}} - R_S^{\text{nadir}}}, \\ z_2 &= \frac{C_S - C_S^{\text{nadir}}}{C_S^{\text{ideal}} - C_S^{\text{nadir}}}, \\ z_3 &= \frac{W_S - W_S^{\text{nadir}}}{W_S^{\text{ideal}} - W_S^{\text{nadir}}}. \end{aligned} \tag{12}$$

5.2. *Mapping the weight vector to the preference value*

Through numerical experiments in this section, we try to make sure that a specific weight vector will result in a corresponding specific preference value, that is, the preference value is a function of the weight vector.

The value function is used to simulate the designer in the solution evaluation process. Assume that the value function takes the form of:

$$V = w_1 z_1 + w_2 z_2 + w_3 z_3, \tag{13}$$

where w_1, w_2, and w_3 are equal to 0.5, 0.3 and 0.2 respectively.

Let $\boldsymbol{\lambda} = [0.40, 0.25, 0.35]$. A genetic algorithm is used to solve the AWTPs model formulated in Equation (3) five times, and the obtained Pareto solutions are evaluated with the value function. The results are listed in Table 1.

It can be concluded from Table 1 that a specific weight vector will result in corresponding specific objective function values and a corresponding preference value. There are still small variations in the obtained preference values because the genetic algorithm may not obtain accurately the same optimal solution in a limited generation number. The ANN model of the designer's preference structure is built with the weight vector acting as the input and the preference value as the desired output. The variations are too small to impact on the function of the ANN model to intelligently guide the multiobjective optimization procedure.

Table 1. Results with respect to a specific weight vector

	Design objectives			*Preference value*
	R_S	C_S	W_S	
1	0.9573	295.36	170.98	2.5274
2	0.9573	295.80	170.95	2.5273
3	0.9573	294.75	170.94	2.5276
4	0.9573	296.12	170.96	2.5273
5	0.9573	296.25	170.97	2.5272

5.3. *Results and discussions*

Value functions are specified in order to simulate the designer in the generated solution evaluation process during the IIMOM procedure. The following five value functions (Sun *et al.*, 2000) are used in order to illustrate the effectiveness of IIMOM to capture different kinds of preference structures. $\mathbf{w} = [0.5, 0.3, 0.2]$ and $KV = 2$ for all the value functions.

1. Linear value function:

$$V = \sum_{i=1}^{3} w_i z_i. \tag{14}$$

2. Quadratic value function:

$$V = KV - \sqrt{\sum_{i=1}^{3} [w_i (1 - z_i)^2]}. \tag{15}$$

3. L_4-metric value function:

$$V = KV - \left[\sum_{i=1}^{3} [w_i (1 - z_i)^4] \right]^{\frac{1}{4}}. \tag{16}$$

4. Tchebycheff metric value function:

$$V = KV - \max_{1 \le i \le 3} \{w_i (1 - z_i)\}. \tag{17}$$

5. Combined value function:

$$V = KV - \left(\sqrt{\sum_{i=1}^{3} [w_i (1 - z_i)^2]} + \max_{1 \le i \le 3} \{w_i (1 - z_i)\} \right) \Big/ 2. \tag{18}$$

The combined value function is obtained by combining the quadratic value function and the Tchebycheff metric value function.

A genetic algorithm is used to solve the AWTP problems, shown in Equation (3), in Step 3 of the IIMOM procedure. Compared with standard nonlinear programming techniques, a genetic algorithm is computationally more expensive, but it has a much better ability to find a global optimum, whereas standard nonlinear programming techniques are easily trapped in local optima. The preference values based on the optimization results obtained by solving these AWTP problems will be used to train the ANN to represent the designer's preferences. In this problem, the population size is chosen to be 100. Decimal encoding is used and the chromosome length is set to be 20, that is, each of the four design variables is represented by a five-digit segment of the chromosome. We use the roulette-wheel selection scheme, a one-point crossover operator with a crossover rate of 0.25, and a uniform mutation operator with a mutation rate of 0.1.

The model parameter vector space reduction factor r is 0.7. Ten Pareto solutions are evaluated at each iteration, in order to make the trained ANN model accurate enough while not requiring too many Pareto solution generating procedures. The Pareto solutions generated in the last five iterations are used to train the ANN model of the designer's preference structure, so that the data used to train the ANN model will focus gradually on the region that the designer is interested in, and make the ANN model more accurate in this region. Except for the first four iterations, in total 50 training pairs are used to train the ANN at each iteration. The numbers 10 and "5" are chosen based on computational experience, and there are no definite criteria on how to choose these numbers.

In this problem, the ANN used to build the model of the designer's preference structure is a three-layered feed-forward neural network, with two neurons in the input layer, one neuron in the output layer, and most often three neurons in the hidden layer. The reason for selecting three hidden neurons is that an ANN with three hidden neurons is believed to be a parsimonious model which can model the nonlinear relationship without overfitting the data when there are two input neurons, one output neuron and, in most cases, 50 training pairs in the training set (Rojas, 1996). Because there are 10 training pairs in the first iteration and 20 training pairs in the second iteration, we use two hidden neurons in these two iterations so that there will not be too many free parameters in the ANN model.

The case of the linear value function is considered first. IIMOM is run for 10 iterations. The weight vector space, average preference value of the generated Pareto solutions at the current iteration, the weight vectors and preference values of $P^{(\text{Best},h)}$ and $P^{(\text{Opt},h)}$ are listed in Table 2.

In Table 2, $[l_1^{(h)}, u_1^{(h)}]$ and $[l_2^{(h)}, u_2^{(h)}]$ are the range of the weights λ_1 and λ_2 at iteration h, $\lambda_3 = 1 - \lambda_1 - \lambda_2$. P^{Best} is the best Pareto solution generated until the current iteration, P^{opt} is the optimization result obtained by solving the optimization problem with the ANN model of the designer's preference structure as the objective function at the current iteration.

The ANN models of the designer's preference structure at iterations 2, 5, 8 and 10 are depicted in Fig. 4(a–d). During the IIMOM process the model parameter vector space is reduced, focusing step-by-step on the part of the Pareto frontier the designer is interested in. The new model

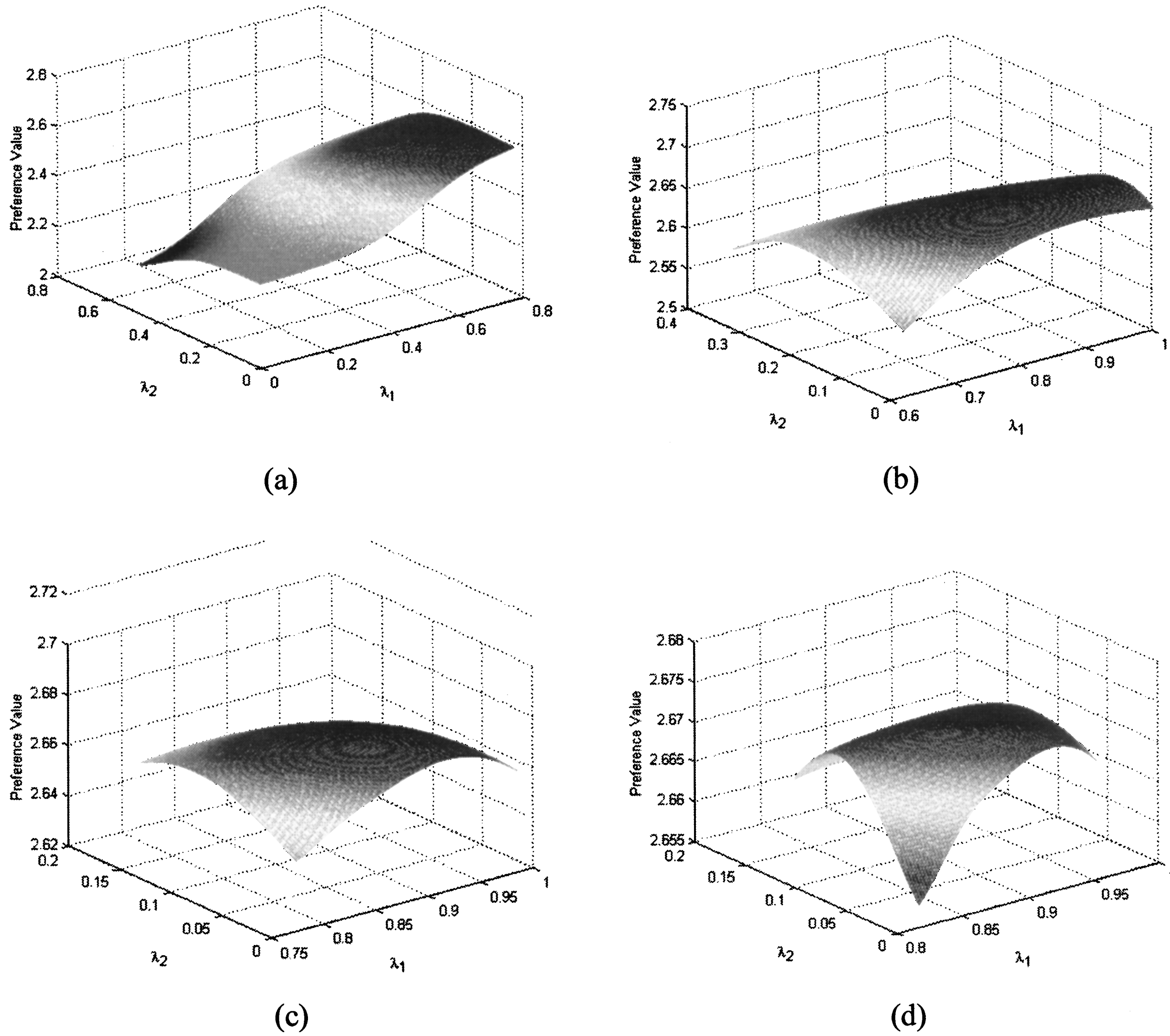

Fig. 4. The ANN model of the designer's preference structure in the case of the linear value function: (a) at the second iteration; (b) at the fifth iteration; (c) at the eighth iteration; and (d) at the 10th iteration.

parameter vector space $\mathbf{\Lambda}^{(h+1)}$ is determined by $P^{(\text{Best},h)}$ and $P^{(\text{Opt},h)}$.

As can be seen from Table 2, the average preference value increases in general during the IIMOM process which means that the model parameter vector space is converging to the model parameter vector that best satisfies the designer's preference. The model parameter vector space is reduced in most iterations, although there might be an iteration in which the model parameter vector space is not reduced. The model parameter vector space is sure to converge in the end. IIMOM captures the hidden tendency of the designer's preference structure through the discrete generated Pareto solutions, and uses all the information the generated Pareto solutions can provide. $P^{(\text{Opt},h)}$ is obtained by solving an optimization problem with the ANN model of the designer's preference structure as the objective. Although the preference value of $P^{(\text{Opt},h)}$ may not be superior to that of $P^{(\text{Best},h)}$, $P^{(\text{Opt},h)}$ does lead the designer to the Pareto frontier part he is interested in. On the other hand, the IAWTPs method presented by Sun *et al.* (2000) only uses the best generated Pareto solution to adjust the weights, and it overlooks some important information provided by other generated Pareto solutions.

The linear value function is used to represent the designer in evaluating the generated Pareto solutions in IIMOM in this case. If the linear value function formulated in Equation (14) is used as the single objective for the reliability optimization problem, the obtained solution must be the Pareto solution that best satisfies the designer's preference. For the purpose of comparison, the Pareto solution obtained in this way, termed the comparing result, is compared with the result obtained using IIMOM in Table 3. It can be seen that the objectives values and the preference value of the result obtained using IIMOM are very close to those of the comparing result. The numerical results illustrate that IIMOM

Table 2. The IIMOM process in the case of a linear value function

h	*Weight vector space*		*Average preference value*	$P^{(Best,h)}$		$P^{(Opt,h)}$	
	$[l_1^{(h)}, u_1^{(h)}]$	$[l_2^{(h)}, u_2^{(h)}]$		*Weight vector* $[\lambda_1, \lambda_2]$	*Preference value*	*Weight vector* $[\lambda_1, \lambda_2]$	*Preference value*
1	[0, 1]	[0, 1]	2.3123	[0.7388, 0.1461]	2.6579	[0, 0]	2.3187
2	[0, 0.7694]	[0, 0.4731]	2.4070	[0.7388, 0.1461]	2.6579	[0.7694, 0.1459]	2.6596
3	[0.4341, 1.0000]	[0, 0.4660]	2.5796	[0.7694, 0.1459]	2.6596	[0.7659, 0.2341]	2.6367
4	[0.5116, 1.0000]	[0, 0.4460]	2.6314	[0.8246, 0.0840]	2.6727	[0.8261, 0.1738]	2.6552
5	[0.6206, 1.0000]	[0, 0.3337]	2.6360	[0.8246, 0.0840]	2.6727	[0.8883, 0.0694]	2.6750
6	[0.6926, 1.0000]	[0, 0.2405]	2.6582	[0.8883, 0.0694]	2.6750	[0.8703, 0.0587]	2.6751
7	[0.7482, 1.0000]	[0, 0.1951]	2.6647	[0.8800, 0.0576]	2.6755	[0.8790, 0.0402]	2.6745
8	[0.7747, 0.9844]	[0, 0.1537]	2.6701	[0.8800, 0.0576]	2.6755	[0.8947, 0.0499]	2.6753
9	[0.8035, 0.9712]	[0, 0.1376]	2.6713	[0.8800, 0.0576]	2.6755	[0.8885, 0.0555]	2.6754
10	[0.8171, 0.9514]	[0, 0.1236]	2.6723	[0.8800, 0.0576]	2.6755	[0.8942, 0.0433]	2.6745

Table 3. The results in the case of the linear value function

	r_1	r_2	r_3	r_4	R_S	C_S	W_S	*Preference value*
IIMOM	0.6394	0.5680	0.6400	0.5551	0.9957	362.9294	183.6814	2.6755
Comparing result	0.6442	0.5680	0.6375	0.5612	0.9959	364.3245	183.8811	2.6756

Table 4. The results in the case of the quadratic value function

	r_1	r_2	r_3	r_4	R_S	C_S	W_S	*Preference value*
IIMOM	0.5726	0.4892	0.5682	0.4813	0.9848	323.4813	176.9803	1.5920
Comparing result	0.5736	0.4900	0.5650	0.4834	0.9849	323.7218	177.0066	1.5920

Table 5. The results in the case of the *L*4-metric value function

	r_1	r_2	r_3	r_4	R_S	C_S	W_S	*Preference value*
IIMOM	0.5403	0.4503	0.5219	0.4365	0.9717	304.8938	173.6443	1.5430
Comparing result	0.5350	0.4484	0.5250	0.4398	0.9719	305.0382	173.5886	1.5430

Table 6. The results in the case of the Tchebycheff metric value function

	r_1	r_2	r_3	r_4	R_S	C_S	W_S	*Preference value*
IIMOM	0.5246	0.4355	0.5090	0.4342	0.9672	300.6453	172.6808	1.8360
Comparing result	0.5214	0.4402	0.5118	0.4298	0.9672	300.5795	172.6974	1.8360

Table 7. The results in the case of the combined value function

	r_1	r_2	r_3	r_4	R_S	C_S	W_S	*Preference value*
IIMOM	0.5615	0.4787	0.5470	0.4698	0.9814	317.3904	175.8632	1.7087
Comparing result	0.5620	0.4760	0.5511	0.4670	0.9812	316.9782	175.8256	1.7087

is effective in capturing the designer's preference structure and obtaining the most satisfying Pareto solution if the designer's preference structure can be approximated as a linear value function.

The IIMOM program we used is based on the MATLAB platform. Here the IIMOM procedure which runs for 10 iterations takes about 10 minutes, that is, each iteration of IIMOM takes about 1 minute. It takes about 5 seconds to solve a single AWTP problem. And it takes about 4 seconds to train the ANN model, and 5 seconds to solve the optimization problem in Equation (4), at each iteration. Generally, for this problem, the IIMOM procedure can be completed in an acceptable time.

In the cases that use either the quadratic value function, the L_4-metric value function, the Tchebycheff metric value function and the combined value function, the optimization results are shown in Tables 4–7. These cases illustrate the same trends as found in the case of the linear value function. The results obtained using IIMOM are very close to those of the comparing results obtained by solving the reliability optimization problems with the corresponding value functions as the single objectives. The results illustrate that IIMOM is effective in capturing different kinds of preference structures of the designer, including linear, quadratic, L_4-metric, Tchebycheff metric and combined modes, and it can obtain the Pareto solution that eventually best satisfies the designer's preference.

5.4. *Discussion on the performances of IIMOM*

1. *Effectiveness in the multiobjective optimization process* (Shin and Ravindran, 1991). IIMOM is very effective in capturing different kinds of preference structures of the designer, and it can obtain the Pareto solution that best satisfies the designer's preference finally.
2. *Ease in actual use*. What the designer needs to do in IIMOM is to evaluate the generated Pareto solutions. Therefore, the designer's cognitive burden is not too heavy, and it is not too complex to use IIMOM in actual problems. How to evaluate the generated solutions and determine their preference values is a key problem.
3. *Convergence*. The ANN model of the designer's preference structure guides the designer to explore the part of the Pareto frontier of interest. The numerical experiments indicate that the IIMOM procedure has a good convergence performance.
4. *Change of the designer's preference structure*. The designer's knowledge of the handled problem increases during the optimization process. After examining some generated solutions, the designer's preference structure may change little by little. IIMOM uses the Pareto solutions generated in the last several iterations to train the ANN model of the designer's preference structure. The training strategy could resolve the problem of the change of the designer's preference structure.

6. Concluding remarks

This work develops an effective multiobjective optimization method, IIMOM, and applies it to the reliability optimization problem of a multi-stage mixed system. In IIMOM, the general concept of the model parameter vector is proposed. From a practical point of view, the designer's preference structure model is built using an ANN with the model parameter vector acting as the input and the preference information articulated by a designer over representative samples from the Pareto set as the desired output. Then with the ANN model of the designer's preference structure as the objective, an optimization problem is solved to search for improved solutions.

Two key advantages of IIMOM are: (i) the ANN model of the designer's preference structure can guide the designer to explore the part of the Pareto frontier of interest efficiently and accurately; and (ii) the improved solutions generated at each iteration are Pareto solutions.

In the reliability optimization problem, five value functions are used to simulate the designer in the solution evaluation process of IIMOM. The results illustrate that IIMOM is very effective in capturing different kinds of preference structures of the designer, and it can obtain the Pareto solution that best satisfies the designer's preference.

Acknowledgements

This research was partially supported by the National Natural Science Foundation of China under contract 50175010, the Excellent Young Teachers Program of the Ministry of Education of China under contract 1766, the National Excellent Doctoral Dissertation Special Foundation of China under contract 200232, and the Natural Sciences and Engineering Research Council of Canada. The constructive suggestions and comments offered by the referees and editors were very much appreciated.

References

Chen, W., Sahai, A., Messac, A. and Sundararaj, G.J. (2000) Exploration of the effectiveness of physical programming in robust design. *Transactions of the ASME. Journal of Mechanical Design*, **122**, 155–163.

Dhingra, A.K. (1992) Optimal apportionment of reliability and redundancy in series systems under multiple objectives. *IEEE Transactions on Reliability*, **41**, 576–582.

Gardiner, L.R. and Steuer, R.E. (1994) Unified interactive multiple objective programming. *European Journal of Operational Research*, **74**, 391–406.

Inagaki, T., Inoue, K. and Akashi, H. (1978) Interactive optimization of system reliability under multiple objectives. *IEEE Transactions on Reliability*, **27**, 264–267.

Korhonen, P., Salo, S. and Steuer, R.E. (1997) Heuristic for estimating nadir criterion values in multiple objective linear programming. *Operations Research*, **45**, 751–757.

Lu, Q.Z., Sheng, Z.H. and Xu, N.R. (1995) A multiobjective decision-making algorithm based on BP-Hopfield neural networks.

Journal of Decision Making and Decision Support Systems, **5**, 93–103.

Park, K.S. (1987) Fuzzy apportionment of system reliability. *IEEE Transactions on Reliability*, **36**, 129–132.

Rao, S.S and Dhingra, A.K. (1992) Reliability and redundancy apportionment using crisp and fuzzy multiobjective optimization approaches. *Reliability Engineering and System Safety*, **37**, 253–261.

Ravi, V., Reddy, P.J. and Zimmermann, H.-J. (2000) Fuzzy global optimization of complex system reliability. *IEEE Transactions on Fuzzy Systems*, **8**, 241–248.

Rojas, R. (1996) *Neural Networks: A System Introduction*. Springer, Berlin, Germany.

Sakawa, M. (1978) Multiobjective optimization by the surrogate worth trade-off method. *IEEE Transactions on Reliability*, **27**, 311–314.

Sakawa, M. (1982) Interactive multiobjective optimization by the sequential proxy optimization technique. *IEEE Transactions on Reliability*, **31**, 461–464.

Shin, W.S. and Ravindran, A. (1991) Interactive multiple objective optimization: survey 1-continuous case. *Computers & Operations Research*, **18**, 97–114.

Stam, A., Sun, M.H. and Haines, M. (1996) Artificial neural network representations for hierarchical preference structures. *Computers & Operations Research*, **23**, 1191–1201.

Sun, M.H., Stam, A. and Steuer, R.E. (1996) Solving multiple objective programming problems using feed-forward artificial neural networks: the interactive FFANN procedure. *Management Science*, **42**, 835–849.

Sun, M.H., Stam, A. and Steuer, R.E. (2000) Interactive multiple objective programming using Tchebycheff programs and artificial neural networks. *Computers & Operations Research*, **27**, 601–620.

Biographies

Hong-Zhong Huang is a Professor in the School of Mechatronics Engineering at University of Electronic Science and Technology of China. He is currently a Visiting Professor at the Department of Mechanical Engineering, University of Alberta, Canada. He received a Ph.D. degree in Reliability Engineering from Shanghai Jiaotong University, Shanghai, China. He has published 120 journal articles and five books in the fields of reliability engineering, optimization design, and fuzzy sets theory. He is a (senior) member of several professional societies, and has served on the boards of professional societies.

Zhigang Tian is a Ph.D. student in the Department of Mechanical Engineering, University of Alberta, Canada. He received a B.S. degree in 2000 and an M.S. degree in 2003 both in Mechanical Engineering from Dalian University of Technology, Dalian, China. His research interests include reliability engineering, optimization, maintenance planning, neural networks, and fault diagnosis.

Ming J. Zuo is a Professor in the Department of Mechanical Engineering at the University of Alberta, Canada. He received a Master of Science degree in 1986 and a Ph.D. degree in 1989 both in Industrial Engineering from Iowa State University, Ames, IA, USA. He worked at the University of Windsor in Canada for a year in 1989. His research interests include system reliability analysis, maintenance planning and optimization, signal processing, and fault diagnosis. He is an Associate Editor of *IEEE Transactions on Reliability* and a member of the Editorial Board of *IIE Transactions on Quality and Reliability Engineering*. He is a senior member of IEEE and IIE.

Contributed by the Reliability Engineering Department