J. of Mult.-Valued Logic & Soft Computing, Vol. 15, pp. 39–50
Reprints available directly from the publisher
Photocopying permitted by license only

GA Based Fuzzy Multi-objective Robust Design

HONG-ZHONG HUANG[1,2], HUANWEI XU[3] AND LIPING HE[3]

[1]*School of Mechatronics Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, 610054, China*
[2]*Chongqing Key Laboratory of Communication Engineering, Chongqing Jiaotong University, Chongqing, 400074, China*
[3]*School of Mechanical Engineering, Dalian University of Technology, Dalian, Liaoning, 116023, China*
E-mail: hzhuang@uestc.edu.cn

Received: October 26, 2007. Accepted: February 20, 2008.

In view of multi-objective optimization problem in robust design, a method combining fuzzy set theory and genetic algorithm is proposed. The approach includes a fuzzy model of multi-objective problem in robust design by taking into account principles of robust design and fuzzy factors in mechanical design, and provides Pareto optimal set to multi-objective robust design using genetic algorithm. Computational results showed that the proposed approach is efficient to multi-objective optimization problem in robust design.

Keywords: Fuzzy set theory, robust design, multi-objective optimization, genetic algorithms.

1 INTRODUCTION

Robust design, originally proposed by Taguchi [1], is an engineering methodology for improving the quality of a product by minimizing the effect of the causes of variation without eliminating these causes [2]. Almost all practical problems in engineering robust design optimization hold the multiple aspects of the objective, while so many methods and approaches have been proposed in the literature to obtain multi-objective robust design. In general these methods can be classified into three types according to preference information.

(1) *Pre-disposal technique* [3,4]. The technique converts multi-objective functions into single objective function through utilizing weighting factors.

These weighting factors imply decision maker's preference information. It is easy to understand and practice this method, but it is difficult to decide the weighting factors. Moreover, the final single objective function can not exactly reflect the relationship between multi-objective functions.

(2) *Interactive technique* [5,6]. Using this kind of methods, decision maker can participate in the whole optimization process. But the correction of optimization solutions depends on subjective estimate of decision maker heavily.

(3) *Post-disposal technique* [7–9]. Comparing with the two foregoing techniques, this kind of methods are capable of producing the efficient set through using the Pareto set concept. Decision maker can select a solution which he/she needs from the efficient set. Among those kind of methods, genetic algorithm is particularly representative. In the literatures, genetic algorithms have been used to solve robust design problems [7–9]. However there are many fuzzy factors in real-world engineering optimization. So a method is not all-around without considering fuzzy factors.

In this paper, we propose a fuzzy model of multi-objective problem in robust design. This model takes into account principles of robust design and fuzzy factors in mechanical design. Pareto optimal set to multi-objective robust design are obtained using genetic algorithm.

The remainder of this paper is presented as follows. In Section 2, a model of multi-objective robust design combining fuzzy techniques is provided. In Section 3, the multi-objective genetic algorithm based on Pareto-optimal set is introduced. Section 4 illustrates our approach through an engineering optimization example, and concluding remarks are given in Section 5.

2 FUZZY MULTIOBJECTIVE ROBUST DESIGN

The quality loss function is first used by Taguchi as a metric for robust optimization. The quality loss function can be expressed as different forms according to different types of quality characteristics ("the nominal the better", "the smaller the better", "the larger the better"). For example, "the nominal the better" type of problem can be expressed as [10]:

$$E[L(Y,T)] = K[\sigma^2 + (\mu - T)^2] \tag{1}$$

where K is the loss function coefficient, T is the target or desired value of the quality characteristic, σ is the variance of the random variable Y, μ is the mean of the objective function. Their relationship can be illustrated in Figure 1.

Design optimization can be divided into two aspects. One is concerned with aligning the peak of the bell shaped response distribution with the target quality (optimizing the mean performance). The other is concerned with making the

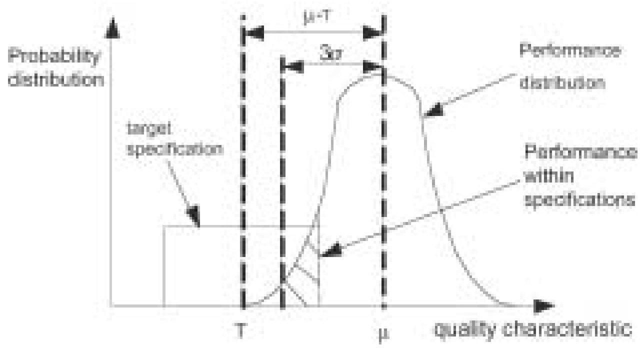

FIGURE 1
Quality distribution in robust design.

bell shaped curve thinner (minimizing the variance σ). The general multi-objective optimization can be expressed as follows:

$$\begin{gathered}\min f_i(x) \quad i = 1, 2, \ldots, N \\ \text{s.t. } g_j(x) \le b_j \quad j = 1, 2, \ldots, J \\ x_L \le x \le x_U\end{gathered} \tag{2}$$

where $f_i(x)$ is the ith objective function, $g_j(x)$ is the jth constraint function, and x_L and x_U indicate the lower and upper bound values of design variables x, respectively.

According to (1), multi-objective robust design can be stated as [11]:

$$\begin{gathered}\min[\mu_i - T, \sigma_i] \quad i = 1, 2, \ldots, N \\ \text{s.t. } g_j(x) + k_j\left(\sum_{i=1}^{n}\left|\frac{\partial g_j}{\partial x_i}\right| \triangle x_i + \sum_{i=1}^{k}\left|\frac{\partial g_j}{\partial x_i}\right| \triangle z_i\right) \le b_j \quad j = 1, 2, \ldots, J \\ x_L + \triangle x \le x \le x_U - \triangle x\end{gathered} \tag{3}$$

where μ_i and σ_i are the mean and standard deviations of the ith objective function $f_i(x)$, respectively. Their values can be obtained through statistical analyses based on simulations or the first-order Taylor expansion if the design deviations of x_i are small. We use the worst case scenario to study the variation of constrains. It is assumed that all variations of system performance may occur simultaneously in the worst possible combination of design variables. The original constraints are modified by adding the penalty terms to each of them, where k_j are penalty factors to be determined by the designer. $\triangle x_i$ is the deviation of the design variable x_i. $\triangle z_i$ is the deviation of the design parameter z_i.

2.1 Construction of the objective membership functions

It is obvious that (3) is a multi-objective optimization problem. The solutions of multi-objective optimization problem have close relationship with all the individual objective functions. It should include contributions of all the individual objective functions. Since the relationship between individual objective functions and multi-objective function is not precisely limited or defined, the solution is biased or dissatisfactory without considering the fuzzy factors [12]. So we use fuzzy set theory to reconstruct the objective functions in (3). Details are indicated in the following step-by-step procedure.

(1) Seek the minimum and maximum values to the individual objective function $f_i(x)$ subjected to the constraints using ordinary optimization procedures. Let the solution be M_i, m_i respectively.

$$\begin{aligned} X &= (x_1, x_2, \ldots, x_n)^T \\ \min f_i(x) &\quad i = 1, 2, \ldots, I \\ \text{s.t. } G_j(x) \le b_j &\quad j = 1, 2, \ldots, J \end{aligned} \tag{4}$$

and

$$\begin{aligned} X &= (x_1, x_2, \ldots, x_n)^T \\ \max f_i(x) &\quad i = 1, 2, \ldots, I \\ \text{s.t. } G_j(x) \le b_j &\quad j = 1, 2, \ldots, J \end{aligned} \tag{5}$$

(2) The membership functions of the fuzzy objective functions are constructed as:

$$\upsilon_{\tilde{f}_i}(x) = \left(\frac{M_i - f_i(x)}{M_i - m_i} \right)^q \tag{6}$$

where $q > 0$ and it can be $\frac{1}{2}, 2, \frac{1}{3}, 3, \ldots$. It is obvious that $\upsilon_{\tilde{f}_i}(x)$ may have different forms. Therefore designer can use it to form different $\upsilon_{\tilde{f}_i}(x)$, so as to reflect characteristics of multi-objective functions, subjective wills of the designer and thereby obtaining more satisfactory solutions.

2.2 Construction of the constraint membership functions

In many optimization problems, the issues of fuzzy bounds about constraints are also exist. For example, the stress induced in a structure may be constrained by an upper bound value as $\sigma(X) \le \sigma^u = 30000$ MPa. This implies that $\sigma = \sigma^u = 30000$ MPa is acceptable, but $\sigma = 30001$ MPa is unacceptable in the ordinary optimization. Since there is no substantive difference between $\sigma = 30000$ MPa and $\sigma = 30001$ MPa, it is more reasonable to assume a transition stage from absolute permission to absolute impermissibility. Thus we stated fuzzy constraints as follows [13]:

$$\upsilon_{\widetilde{G}_j}(x) = \begin{cases} 0, & \text{if } G_j(x) > b_j + d_j \\ 1 - \left\{ \dfrac{G_i(x) - b_j}{d_j} \right\}, & \text{if } b_j \le G_j(x) \le b_j + d_j \\ 1, & \text{if } G_j(x) < b_j \end{cases} \tag{7}$$

where d_j denotes the permissible variation of G_j, $j = 1, 2, \ldots, J$. Certainly the constraint membership function can be different forms according to request of design and properties of constraints.

By considering the optimum solution as the intersection of the membership functions of the objective functions and constraints, the solution of the fuzzy multi-objective optimization problem can be found by determining λ_i and x

$$\begin{aligned} &\max \lambda_i \quad i = 1, 2, \ldots, N \\ &\text{s.t. } \lambda_i \le \upsilon_{\widetilde{f}_i}(x), \\ &\lambda_i \le \upsilon_{\widetilde{G}_j}(x), \quad j = 1, 2, \ldots, J \end{aligned} \tag{8}$$

3 MULTIOBJECTIVE GENETIC ALGORITHMS

From a mathematical point of view, genetic algorithms (GA's) are categorized as random walk search methods with direction exploitation [7]. They have been utilized in a broad spectrum of engineering application. Pareto GA which based on the concept of Pareto set is especially well-suited for identifying candidate non-inferior designs of multi-objective optimization. In the Pareto GA three techniques are added [14]:

(1) *Ranks of candidate designs based on the concept of Pareto domination.*

(2) *Sharing technique.*

(3) *Pareto set filter.*

3.1 Ranks of candidate designs

In order to distinguish the non-inferior solutions from population, the technique to rank candidate designs is adopted. At the beginning of selection, each individual in the population of design is assigned a rank equal to the degree of Pareto domination. The degree of domination of an individual design is the total number of designs in the population that dominate that design. A design is said to dominate another in the population if it is at least equal in all objectives to that individual and better in at least one. Non-dominated designs, or those that are not dominated by any individuals in the population, are assigned a rank of one. In Figure 2, where the objectives are to minimize both f_1 and f_2, an example of a Pareto domination ranked population is shown. The fitness value of each individual is the reciprocal of its rank value.

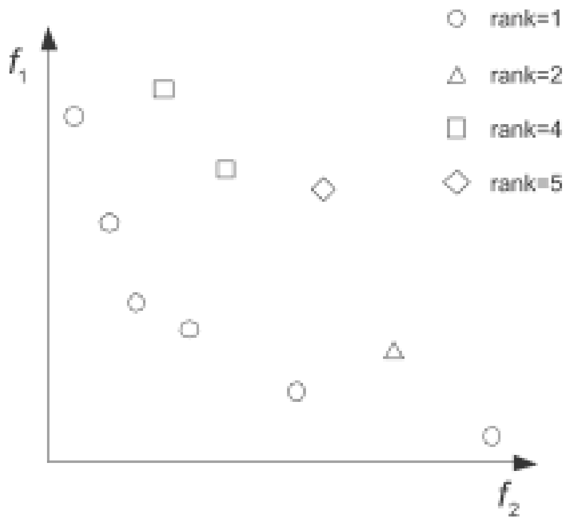

FIGURE 2
Pareto domination ranking for a population.

3.2 Sharing procedure

To prevent the genetic drift phenomenon, a form of sharing should be carried out when there is no preference between two candidates. This form of sharing maintains the genetic diversity along the population fronts and allows the GA to develop a reasonable representation of the Pareto-optimal front. Generally the basic idea behind sharing is that the more individuals are located in the neighborhood of a certain individual, the more its fitness value is degraded. The definition of sharing function is as follows:

$$f_{\text{share}}(X, Y) = \begin{cases} 0 & d_{XY} \geq \sigma_{\text{share}} \\ 1 - d_{XY}/\sigma_{\text{share}} & d_{XY} < \sigma_{\text{share}} \end{cases} \tag{9}$$

where d_{XY} is a normalize Euclidean distance between individual i and another individual j in the current population, σ_{share} is a prespecified distance value.

For the individuals crowed in the process of evolution, we degrade their fitness values by adding sharing function on them as

$$\delta(X)_{\text{share}} = \frac{n\delta(X)}{\sum_{i=1}^{n} f_{\text{share}}(X, Z_i)} \tag{10}$$

where n is the size of the population. X is an individual in the population. Z_i is another individual in the population which is different from$X$. $\delta(X)$, $\delta(X)_{\text{share}}$ are fitness values before and after adding sharing function.

3.3 Pareto set filter

In some problems, the Pareto-optimal set can be extremely large or even an infinite number of solutions. In this case, reducing the set of non-dominated solutions without destroying the characteristics of the trade-off front is desirable from the decision maker's point of view. A Pareto set filter is employed to reduce the Pareto set to manageable size. It can be described as follows. In each evolving generation, we reserve the individuals whose rank values are 1 in Pareto set filter. If the Pareto set filter exceeds the maximum allowable size, we rank these individuals and eliminate dominated individuals. If the Pareto set filter still exceeds the maximum allowable size, we use sharing procedure to eliminate some individuals who are located in a neighborhood of a certain individual.

3.4 Disposal of mixed discrete variables [15]

The three kinds of design variables are handled using the following mapping functions:

(1) *Discrete variables with equal spacing.*

$$x_i = x_i^{\mathrm{L}} + (N_i - 1)\Delta \mathrm{s}, \quad i = 1, \ldots, q \tag{11}$$

(2) *Discrete variables with unequal spacing.*

$$x_i = p_{N_i, i-q}, \quad q < i \leq d \tag{12}$$

(3) *Continuous variables.*

$$x_i = x_i^{\mathrm{L}} + (N_i - 1)\varepsilon_i, \quad d < i \leq n \tag{13}$$

where x_i denotes the ith design variable, x_i^{L} indicates the lower bound value of design variable x_i. N_i is the natural number corresponding to x_i. Δs is the value of equal spacing. q represents the matrix of the values of discrete variables with unequal spacing. The value of ε_i depends on the requirement of engineering precision. Corresponding to each variable x_i, only one value of N_i exists. Thus, the problem of finding the optimal design variables can be transformed into that of finding the optimal values of N_i. Thereupon, all operations of the iterative procedure in genetic algorithm are to determine suitable values of N_i, which in turn can be used to obtain the physical values of the design variables.

3.5 Genetic operators

Each design variable N_i is encoded as a finite length binary digit string. These strings represent artificial chromosomes. Some pairs of strings are randomly selected as parents to reproduce offspring according to the selection rule related to their fitness values. For this case, the chromosomes are sorted according to their fitness. In the crossover operator, the multi-point crossover strategy is selected. It can be stated as follows: multi-points are randomly chosen, which cut the binary strings of parents into several segments. Some segments of father string can be exchanged with those of mother string. An example is as follows:

Parents		Offspring
$01\underline{01}00\underline{10}$	Crossover →	$01\underline{10}00\underline{01}$
$10\underline{10}10\underline{01}$		$10\underline{01}10\underline{10}$

Mutation is the occasional random alteration on a bit-by-bit basis. Similar to the crossover, multi-point random mutation proved to be the best choice in this step. An example is: $11\underline{00}10\underline{01} \Rightarrow 11\underline{11}10\underline{10}$.

Constraints can be applied explicitly in genetic algorithms by adding variable penalty functions that increase the values of the objective functions proportionally to the magnitude of constraint violations.

4 ILLUSTRATIVE EXAMPLE

To demonstrate our fuzzy robust optimization method, we applied it to a well-know welded problem originally formulated by Ragsdell and Phillips [16]. We slightly modified the problem by assuming that there are variations in one design variable and one design parameter.

The original formulation of the problem is shown as

Find $x = [x_1, x_2, x_3, x_4]^{\mathrm{T}} = [t, b, h, l]^{\mathrm{T}}$

Min $\mathrm{F} = (1 + c_1)x_3^2 x_4 + c_2 x_1 x_2 (L + x_4)$

s.t.

$$g_1(x) = [(\tau')^2 + 2\tau'\tau'' \cos\theta + (\tau'')^2]^{1/2} - \tau_{\mathrm{d}} \le 0$$

$$g_2(x) = 6FL/(x_2 x_1^2) - \sigma_{\mathrm{d}} \le 0$$

$$g_3(x) = F - \frac{4.013\sqrt{EI\alpha}}{L^2}\left[1 - \frac{x_1}{2L}\sqrt{\frac{EI}{\alpha}}\right] \le 0$$

$$g_4(x) = 4FL^3/(Ex_1^3 x_2) - 0.25 \le 0$$

$$g_5(x) = x_3 - x_2 \le 0$$

$$g_6(x) = 0.125 - x_3 \le 0$$

$$0.5 \le x_1 \le 10, \quad 0.5 \le x_2 \le 2$$

$$0.1 \le x_3 \le 2, \quad 1 \le x_4 \le 10$$

x_1, x_2 are discrete variables with equal spacing, respectively. The value of spacing is 0.5. x_3 is a continuous variable, whose engineering precision is 0.01. x_4 is an integer variable. $\Delta x_3 = 0.1$ and $\Delta L = 1$. For the optimal design of the problem in a fuzzy environment, an allowable deviation of 10% for each design constraint is considered as a fuzzy transition zone. At first we convert the problem into conventional bi-objective robust design, then we apply our method to transform the model of conventional robust design into the model of fuzzy multi-objective robust design. Finally we implement Pareto GA in Matlab6.0 to solve the problem. In the Pareto GA, the control parameters are as follows. The size of population is 60, the maximal evolving generation is 200, crossover rate is 0.9, mutation rate is 0.1 and the size of Pareto set filter is 100. The values of these parameters are set with reference to ref. [14].

The results are shown in Figure 3. In this figure, the Pareto fronts of both conventional and fuzzy robust design solutions which are solved by using Pareto GA are given. Figure 3 indicates that Pareto GA can generate a majority of Pareto optimal solutions. It is also evident that the objective function is in conflict with the variance of the objective function. Two objectives, which are simultaneously minimized, can never be attained. The Pareto fronts also illustrate that the conventional robust Pareto set is inferior to the fuzzy robust Pareto set. In Figure 4, we give the fuzzy robust design solved by Weighted-sum method (WS method). In WS method, w_1 evenly increases from 0 to 1 while w_2 decreases from 1 to 0 accordingly. From the result, we can observe that although w_1 and w_2 are changed evenly, the efficient solutions obtained by using the WS method are not distributed evenly in the graph. There is an unequal gap between any two solutions. The results illustrate that it is difficult for design maker to obtain a satisfying solution through the use of WS method.

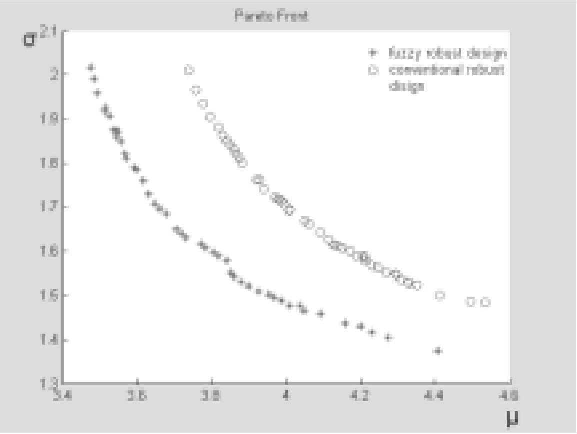

FIGURE 3
Pareto front comparisons between conventional and fuzzy robust design.

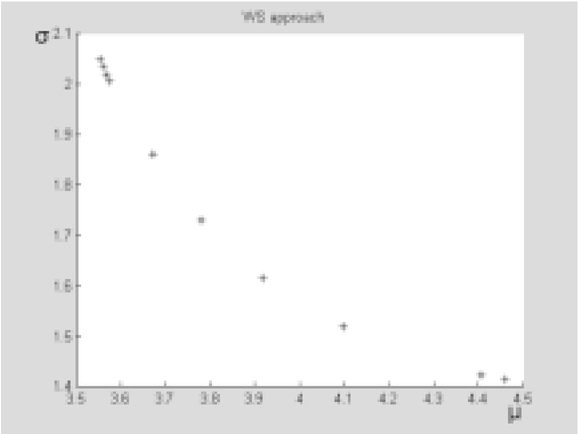

FIGURE 4
Efficient solutions using the WS approach.

However by using Pareto GA, design maker can select a satisfying solution easily from the Pareto front. The results we obtained for this engineering problem clearly proves the advantage of our approach over the conventional robust design and WS method.

5 CONCLUSIONS

A synthetic optimization method that combines fuzzy set theory with Pareto genetic algorithm to perform robust design is proposed in the paper. This method can compromise between multi-objective functions, and find globally compromise solutions for fuzzy multi-objective robust design optimization problems containing mixed-discrete design variables. As a demonstration, we applied the method to an engineering example. In the example we observed that the solutions obtained are more reliable and satisfactory than conventional robust design. However, design maker has to select a solution from the Pareto front by using our method. It may be difficult for design maker to make such a decision in complicated engineering optimization problems. This difficulty will be investigated as part of our future research in this area.

ACKNOWLEDGMENTS

This research was partially supported by the National Natural Science Foundation of China under the contract number 50775026, the National High Technology Research and Development Program of China (863 Program) under the contract number 2007AA04Z403, the Specialized Research Fund for the Doctoral Program of Higher Education of China under the contract number 20060614016, and the Open Foundation of Chongqing Key Laboratory of Communication Engineering.

REFERENCES

[1] Taguchi G. Quality engineering through design optimization. *White Plains*, Krauss International Press, New York, 1986.

[2] Phadke M. S. *Quality Engineering Using Robust Design*. Prentice Hall, New Jersey, 1989.

[3] Sundaresan S., Ishii K. and Houser D. A robust optimization procedure with variations on design variables and constraints. *Proceedings of ASME 1993 Advances in Design Automation Conference* **69**(1) (1993), 379–386.

[4] Hazelrigg G. A. *Systems Engineering and Approach to Information-Based Design*. Prentice Hall, New York, 1996.

[5] Chen W., Sahai A., Messac A. and Sundararaj G. J. Exploration of the effectiveness of physical programming in robust design. *ASME J Mech Des* **122**(2) (2000), 155–163.

[6] Messac A. and Ismail Y. A. Multiobjective robust design using physical programming. *Struct Multidisc Optim* **23** (2002), 357–371.

[7] Forouraghi B. A genetic algorithm for multiobjective robust design. *Appl Intell* **12** (2000), 151–161.

[8] Kurt H. and Kemper L. Robust design through the use of a hybrid genetic algorithm. *Proceedings of ASME 2002 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2002.

[9] Zhu X., Wang A. and Zhang H. Robust design for mechanical/structural system by using non-stationary penalty function with GA. *J Mech Sci Technol* **19**(1) (2000), 49–51.

[10] Yu J. C. and Ishii K. Design for robustness based on manufacturing variation patterns. *ASME J Mech Design* **120** (1998), 196–202.

[11] Chen W., Allen J. K., Tsui K. L. and Mistree F. A procedure for robust design: minimizing variations caused by noise factors and control factors. *ASME J Mech Des* **118** (1996), 478–485.

[12] Chen W., Wiecek M. M. and Zhang J. Quality utility-a compromise programming approach to robust design. *ASME J Mech Des* **121**(2) (1998), 179–187.

[13] Huang H. Z. *Fuzzy Design*. China Machine Press, Beijing, 1999.

[14] Rao S. S. Multi-objective optimization of fuzzy structural systems. *Int J Numer Meth Eng* **124** (1987), 1157–1171.

[15] Wang X. Pareto genetic algorithm for multi-objective optimization design. *J Syst EngElectron* **25**(12) (2003), 1558–1561.

[16] Xiong Y. and Rao S. S. Fuzzy nonlinear programming for mixed-discrete design optimization through hybrid genetic algorithm. *Fuzzy Set Syst* **146** (2004), 167–186.

[17] Ragsdell K. M. and Phillips D. T. Optimal design of a class of welded structures using geometric programming. *ASME J Eng Ind* **98**(3) (1976), 1021–1025.

BIOGRAPHY

Dr. Hong-Zhong Huang is a full professor and the Dean of the School of Mechanical, Electronic, and Industrial Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, China. He has held visiting appointments at several universities in the USA, Canada, and Asia. He received a Ph. D. degree in Reliability Engineering from Shanghai Jiaotong University, China. He has published 120 journal papers and 5 books in fields of reliability engineering, optimization design, fuzzy sets theory, and product development. He is an Editorial Board Member of International Journal of Reliability, Quality and Safety Engineering, International Journal of Reliability and Applications, International Journal of Performability Engineering, Advances in Fuzzy Sets and Systems, and The Open Mechanical Engineering Journal. He received the Golomski Award from the Institute of Industrial Engineers in 2006. His current research interests include system reliability analysis, warranty, maintenance planning and optimization, computational intelligence in product design.