AIAA JOURNAL
Vol. 43, No. 7, July 2005

Mixed-Discrete Fuzzy Multiobjective Programming for Engineering Optimization Using Hybrid Genetic Algorithm

Singiresu S. Rao* and Ying Xiong†
University of Miami, Coral Gables, Florida 33124-0624

Although much attention has been focused on the development and applications of fuzzy optimization, multi-objective programming, and mixed-discrete optimization methods separately, fuzzy multiobjective optimization problems in mixed-discrete design space have not been addressed in the literature. It is mainly because of the lack of mature and robust theories of mixed-discrete and multiobjective optimization. In most practical applications, designers often encounter problems involving imprecise or fuzzy information, multiple objectives, and mixed-discrete design variables. A new method is presented in which the fuzzy λ formulation and game theory techniques are combined with a mixed-discrete hybrid genetic algorithm for solving mixed-dixcrete fuzzy multiobjective programming problems. Three example problems, dealing with the optimal designs of a two-bar truss, a conical convective spine, and a 25-bar truss, demonstrate that the method can be flexibly and effectively applied to various kinds of engineering design problems to obtain more realistic and satisfactory results in an imprecise environment.

Nomenclature

D	=	fuzzy feasible solution domain
$\boldsymbol{d}$	=	favorable search direction
$\mathrm{Fit}_i(X)$	=	fitness function
$f_i(X)$	=	ith objective function
$f_i^{\max}$	=	upper bound of ith objective function
$f_i^{\min}$	=	lower bound of ith objective function
G_j	=	allowable interval of the constraint function g_j
$g_j(X)$	=	jth inequality constraint function
k	=	number of objective functions
M	=	size of the population
m	=	number of inequality constraints
n	=	total number of design variables
nd	=	number of discrete design variables
nq	=	number of discrete design variables with equal spacing
X	=	design vector
X_c	=	centroid of the composite polygon
X_w	=	vertex with the lowest fitness value
x_i	=	ith design variable
$\mu_D(X)$	=	membership function of the design vector
$\mu_{\mathrm{fi}}(X)$	=	membership function of the ith objective
$\mu_{\mathrm{gj}}(X)$	=	membership function of the jth constraint

I. Introduction

THE multiobjective optimization problems were originally investigated in the field of mathematical economics. The earliest reported in-depth work appears to be that of Kuhn and Tucker.[1] Since then, a variety of techniques and applications of multiobjective optimization have been developed. Because no unique solution exists that would be optimum for all of the individual objective functions simultaneously, a concept known as Pareto optimality, which is different from the optimality concept used in scalar optimization, has been used in most of the available multiobjective optimization methods.

Most design problems, traditionally, are stated in precise mathematical forms. However, it is recognized that most phenomena encountered by designers and decision makers would take place in a fuzzy environment in which the statements might be uncertain, vague, fuzzy, or imprecise. (The terms *uncertain*, *vague*, *fuzzy*, and *imprecise* are used to mean the same in the context of this paper.) This might be caused by different reasons: designers might not be able to express their objectives precisely because their utility functions are not definable precisely, or the phenomena of the engineering problem might be described only in a fuzzy way. For instance, in control system design the forcing frequency is expected to be "substantially away" from the natural frequency of the system, and in the thermal design of buildings, the design temperature or "comfortable temperature" is subjective and is not precisely defined. Generally uncertain variables or quantities in optimization problems are described using terms such as high, large, essentially, and roughly. All of these terms represent fuzzy information. Usually it is difficult to describe the goals and constraints of optimization problems related to such systems by crisp relations through equations and/or descriptions. It is possible that a small violation of a given constraint can lead to a more efficient and practical solution of the problem. So it is more reasonable that there should be transition stages from absolute permissibility to absolute impermissibility when the allowable interval of a physical variable is determined, that is, the ordinary subset should be replaced by a fuzzy subset along the real axis. The fuzzy set formulation represents, in fact, a subjective estimation of a possible effect of a given result on the objective function and constraints. From this point of view, a fuzzy set provides a model to express fuzzy relationships as just described and permits the incorporation of vagueness in the conventional set theory, which can be used to deal with uncertainty quantitatively. It was proven to be an efficient tool, and the mathematical developments have provided the theoretical basis necessary for use in practical applications.

The fuzzy set theory was initiated by Zadeh[2] in 1965, and the concept of fuzzy optimization was first introduced by Bellman and Zadeh[3] in their seminal paper on decision making in a fuzzy environment, in which the concepts of fuzzy constraint, fuzzy objective, and fuzzy decision were introduced. These concepts were subsequently profusely used by most investigators. Fuzzy optimization is a flexible approach that permits a more adequate solution of real problems in the presence of vague information. In the last two decades the principles of fuzzy optimization were critically studied, and the technologies and solution procedures have been investigated within the scope of fuzzy sets. Today, similar to the developments in crisp optimization, different kinds of mathematical models have been proposed, and many practical applications have been implemented to solve fuzzy optimization problems in the various engineering fields,

---

Received 13 June 2003; revision received 22 October 2004; accepted for publication 9 November 2004.

*Professor and Chairman, Department of Mechanical Engineering. Associate Fellow AIAA.
†Postdoctoral Fellow, Department of Mechanical Engineering.

such as mechanical design and manufacturing,[4–6] power systems,[7,8] water resources research,[9,10] and control systems.[11–13]

Recently, much attention has been paid in developing fuzzy programming techniques for multiobjective optimization problems. Buckley[14] presented a method of using fuzzy programming, through the use of either the min or the product operator, which can be used to generate the whole Pareto optimal set for nonlinear concave (or convex) multiobjective programming problems. Kassem[15] researched multiobjective nonlinear programming (MONLP) problems with fuzzy parameters in the objective functions using the concept of α-pareto optimality without differentiability. The stability of multiobjective nonlinear programming problems with fuzzy parameters in the objective and constraint functions was discussed in Refs. 16 and 17. Mohan and Nguyen[18] elaborated an interactive satisfying method used to solve linear as well as a class of nonlinear multiobjective problems in mixed fuzzy-stochastic environment involving various kinds of uncertainties related to fuzziness and/or randomness.

Several studies on the optimization of uncertain engineering systems using different approaches have been reported in the literature. For example, Refs. 19 and 20 modeled the uncertainty using probabilistic/reliability approaches in the optimization of engineering systems. A physical programming approach was used by Messac and his associates in Refs. 21 and 22. An evidence-based uncertainty model was used by Chen and Rao[23] for the multicriteria optimization of mechanical systems. Rao and Dhingra[24] considered the problem of reliability allocation and redundancy apportionment for multistage systems with components having time-dependent reliability, using crisp and fuzzy MONLP optimization approaches coupled with heuristic procedures. A cooperative fuzzy game theoretic approach to multiple objective design optimization was also proposed by them.[25] Several computational models, including simple additive, weighted additive, and preemptive priority models, were given in a fuzzy nonlinear goal programming approach, and the methodologies were illustrated with the help of two structural optimization problems involving multiple goals.[26] Further, an attempt was made in Ref. 27 to apply genetic algorithms and goal programming to solve problems involving multiple objectives and imprecise information.

Sakawa has conducted research in the field of multiobjective fuzzy nonlinear programming. He investigated various fuzzy methods for solving multiobjective nonlinear programming problems, including 1) fuzzy dual decomposition method,[28] by formulating the dual problem of the original problem based on dual decomposition technique, and deriving a compromising solution of the decision maker based on a fuzzy decision; 2) primal decomposition method,[29] by introducing a right-hand-side allocation vector and a two-level optimization algorithm to derive a satisfying solution; 3) interactive fuzzy satisfying method,[30–33] by eliciting corresponding membership functions through interaction with the decision maker, considering current membership values as well as tradeoff rates, updating reference membership values, and deriving a satisfying solution from among a Pareto-optimal solution set, and the combination of interactive fuzzy satisfying method and floating-point genetic algorithm[34] was also proposed to solve multiobjective nonconvex programming problems.

Although several efforts have been made in the development and applications of multiobjective optimization techniques in a fuzzy environment, fuzzy multiobjective optimization problems in mixed-discrete design space were not attempted in the literature. It is mainly because of the lack of mature and robust theories of mixed-discrete optimization that can be coupled with fuzzy multiobjective optimization techniques. However, in real-life projects decision makers are often forced to face such kind of design problems. It is, therefore, practical to appropriately develop a robust and effective method that can integrate reliable mixed-discrete and multiobjective optimization algorithms into a suitable fuzzy programming technique.

In this work, a new programming method is presented, in which the fuzzy λ formulation coupled with the game theory technique is combined with a new mixed-discrete hybrid genetic algorithm (MDHGA) for solving mixed-discrete fuzzy multiobjective programming (MDFMP) problems.

II. MDFMP Approach

A multiobjective optimization problem in the mixed-discrete design space can be stated as follows:

Find X, which minimizes

$$f(X) = [f_1(X), f_2(X), \ldots, f_k(X)]^T \tag{1}$$

subject to

$$g_j(X) \le b_j, \qquad j = 1, 2, \ldots, m$$

$$X = [x_1, \ldots, x_{\mathrm{nq}}, \ldots, x_{\mathrm{nd}}, \ldots, x_n]^T$$

$$x_i^L \le x_i \le x_i^U, \qquad i = 1, 2, \ldots, n$$

When expressed in a fuzzy environment, this problem can be formulated as follows:

Find X, which minimizes

$$f(X) = [f_1(X), f_2(X), \ldots, f_k(X)]^T \tag{2}$$

subject to

$$g_j(X) \in G_j, \qquad j = 1, 2, \ldots, m$$

$$X = [x_1, \ldots, x_{\mathrm{nq}}, \ldots, x_{\mathrm{nd}}, \ldots, x_n]^T$$

$$x_i^L \le x_i \le x_i^U, \qquad i = 1, 2, \ldots, n$$

If both the objectives and constraints are fuzzy, a fuzzy multiobjective optimization problem can be scalarized so that a fuzzy single objective optimization approach with mixed-discrete variables can be used for its solution. The procedure consists of the following two steps:

In the first step, suitable membership functions are determined for each of the objectives and constraints, such that the degree of satisfaction of the objectives and constraints can be measured in a quantitative manner. In this work, the construction of membership functions is based on the techniques of game theory.

In the second step, a single-objective mathematical model is generated for crisp mixed-discrete optimization. By defining a fuzzy feasible solution domain D corresponding to the objective functions and the constraints as

$$D = \left\{ \bigcap_{i=1}^{k} \mu_{\mathrm{fi}}(X) \right\} \cap \left\{ \bigcap_{j=1}^{m} \mu_{\mathrm{gj}}(X) \right\} \tag{3}$$

the membership function of a design vector is given by

$$\mu_D(X) = \min_{i,j} \{\mu_{\mathrm{fi}}(X), \mu_{\mathrm{gj}}(X)\} \tag{4}$$

A design vector X is said to be feasible if $\mu_D(X) > 0$. The membership function $\mu_D(X)$ describes an overall degree of satisfaction of any design vector to all of the objective functions and constraints in the fuzzy feasible domain D. The optimum solution is selected to yield the maximum value of μ_D such that

$$\mu_D(X^*) = \max \mu_D X = \max\{\min[\mu_{\mathrm{fi}}(X), \mu_{\mathrm{gj}}(X)]\} \tag{5}$$

This max-min problem can then be solved by using the fuzzy λ-formulation technique[35] that can be stated mathematically as follows:

Minimize

$$-\lambda$$

subject to

$$\lambda \le \mu_{\mathrm{fi}}(X), \qquad i = 1, \ldots, k$$

$$\lambda \le \mu_{\mathrm{gj}}(X), \qquad j = 1, \ldots, m \tag{6}$$

A. Membership Functions

The selection of the membership functions of objectives and constraints in multiobjective programming is based on engineering judgment and physical insights of the problem as well as the needs of the design. Different shapes of objective membership functions formulated with different sets of $(f^{\min}, f^{\max})$ usually yield different results. Thus it is very important to determine the appropriate functions of $\mu_{\mathrm{fi}}(X)$. To construct $\mu_{\mathrm{fi}}(X)$, a lower bound $f_i^{\min}$ and an upper bound $f_i^{\max}$ of each objective should be determined. Reference 36 gives more illustrations on the construction of the mixed-discrete objective membership functions and concludes that the objective membership function composed of $(f_d^{\min}, f_d^{\max})$ that are obtained in the discrete design space is often the best choice. But there are still several options for the value of $f_d^{\max}$: it can be represented by the absolute maximum value of the corresponding objective in discrete design space, or it can be represented by the relative maximum value of the objective when referencing the optimum solutions of other objectives. In this work, the construction of $\mu_{\mathrm{fi}}(X)$ is based on the cooperative game theory technique, which can be described as follows:

1) Find the solutions of the individual single-objective optimization problems in discrete design space.
2) Determine the best and worst solutions possible for each of the objective functions.
3) Use these solutions as boundaries of fuzzy ranges in the corresponding fuzzy optimization problems.
4) Construct fuzzy objective membership functions using a linear expression.

B. Mixed-Discrete Hybrid Genetic Algorithm

Genetic algorithms (GAs) are recently developed optimization techniques that were originally introduced by Holland[37] in 1975. GAs are heuristic combinatorial search techniques based on the concepts of natural genetics and Darwinian survival of the fittest. Since their introduction as a method of optimization, GAs have been efficiently applied to commerce, engineering, mathematics, medicine, and pattern recognition with promising results.

In GAs, design variables are usually coded into binary strings. Starting with a fixed-size subset of solutions called the population, a sequence of operations, simulating a natural evolution process and consisting of selection, crossover, and mutation, is applied to the population to generate a new one with higher fitness value, which means that better designs are obtained. Such a sequence of operations constitutes what is called a generation. Generations are then applied repeatedly until some stopping criterion is met. A number of studies have been made to establish not only an intuitive convergence of the method, but also a rigorous mathematical convergence proof.[38] The flexibility, globality, parallelism, simplicity, versatility, and good problem solving capability are attractive advantages that make GAs very useful and successful in the mixed-discrete engineering optimization field.

GAs are a family of algorithms that have the same basic structure and differ from one another with respect to the strategies and parameters used to control the search. Often the choice of parameters, such as population size, crossover probability and mutation probability, can have a significant impact on the effectiveness of GAs, that is, each combination of the parameters can result in a different optimum solution. It is a troublesome task to tune these adjustable parameters if we want to obtain acceptable solutions. Another disadvantage of GAs is their high computational cost, especially with the finite element analysis of large-scale problems, in which the number of function evaluations required for a global optimum result is very large.

To overcome these shortcomings of traditional GAs, a MDHGA is proposed in this paper for solving mixed-discrete optimization problems. This algorithm not only possesses all features of the traditional GAs but also has some distinguishable characteristics and advantages in terms of design variables, crossover, mutation, and regeneration operations.

1. *Treatment of Design Variables*

Three kinds of design variables involved in this work are handled using the following mapping functions:

1) Discrete variables with equal spacing

$$x_i = x_i^L + (N_i - 1)\, dsp_i, \qquad i = 1, \ldots, \mathrm{nq} \tag{7}$$

2) Discrete variables with unequal spacing

$$x_i = q_{N_i,\mathrm{i-nq}}, \qquad \mathrm{nq} < i \le \mathrm{nd} \tag{8}$$

3) Continuous variables

$$x_i = x_i^L + (N_i - 1)\varepsilon_i, \qquad \mathrm{nd} < i \le nn \tag{9}$$

where nn is the number of design variables, N_i is the natural number corresponding to x_i, dsp_i is the characteristics of discrete variable, defined as

$$dsp_i = \begin{cases} \text{discrete increment of the } i\text{th discrete} & \\ \quad \text{variable with equal spacing,} & 1 \le i \le \mathrm{nq} \\ \text{discrete number of the } i\text{th discrete} & \\ \quad \text{variable with unequal spacing,} & \mathrm{nq} \le i \le \mathrm{nd} \end{cases}$$

q represents the matrix of values of discrete variables with unequal spacing, of dimension $w^*(\mathrm{nd} - \mathrm{nq})$, and w is the maximum permissible amount of discrete values among all discrete variables with unequal spacing.

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 are to determine suitable values of N_i, which in turn can be used to obtain the physical values of the design variables.

2. *MDHGA Scheme*

The mixed-discrete hybrid genetic algorithm can be summarized by the following steps:

1) Binary encoding of the problem: Each design variable N_i is expressed as a finite length binary digit string. These strings represent artificial chromosomes and every digit in the string is an artificial gene.

2) Initialization of population: The size of the population M is predefined and fixed throughout. The initial population is created randomly within the feasible design space.

3) Evaluation of population: The fitness is a quality value that is a measure of the reproductive efficiency of living creatures according to the principle of survival of the fittest. In the genetic algorithm, the fitness function is chosen as a measure of goodness to be maximized. The fitness function used to evaluate individual is chosen as

$$\phi_j[X] = f_j[X] + R\sum_{i=1}^{m} \max\{0, g_{i,j}\}, \qquad j = 1, \ldots, M$$

$$\mathrm{Fit}_j[X] = 1/|\phi_j[X] - C|, \qquad j = 1, \ldots, M \tag{10}$$

where R is a large enough positive constant (whose value is assumed at the beginning of the computations) and C is a small enough negative constant to make the fitness function large enough during computations. The larger the fitness value, the better the individual.

4) Selection: Some pairs of strings are randomly selected as parents to reproduce offsprings based on the selection rule. For this, the candidates are sorted according to their fitness. The crossover rate is predefined such that a fixed proportion of the population is randomly generated. Each random number is transformed into an ordinal number of the ranked candidates according to Eq. (11), and the corresponding string is selected as crossover parents of the next generation:

$$p1 = (ps - 1)\cdot r^{1.5} + 0.5, \qquad p1 = \langle p1 \rangle \tag{11}$$

where ps is the size of population, r the random number, $p1$ is the ordinal number, and the operator $\langle\,\rangle$ denotes the integerization of $p1$.

5) Crossover: Three different operators are to be considered. The first one can be called the single-point crossover strategy in which the binary strings of two parents are to be cut at a randomly chosen position and two offsprings are reproduced. The first of them inherits the previous part of mother's and latter part of father's binary string, and the second is exactly the reverse. The following example illustrates the procedure:

Parents	Offspring
0101 | 0010	0101 | 1001
1010 | 1001	1010 | 0010

The second strategy is called the two-point crossover that involves randomly choosing two crossover sites in the binary strings of parents and exchanging the digits between the two sites. An example of this strategy is given next:

Parents	Offspring
01 | 0100 | 10	01 | 1010 | 10
10 | 1010 | 01	10 | 0100 | 01

The last operator is referred to as the random multipoint crossover strategy in which more 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 given here:

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

In practical applications, it was found that when the population size is large enough ($n \geq 100$), there is no notable difference in the efficiencies of the solutions among these three strategies. If the population size is comparatively small, the multipoint crossover has the highest efficiency, followed by the two-point crossover, with the one-point crossover having the lowest efficiency. Figure 1 shows a comparison of their efficiencies with respect to the population size, derived from the numerical experimentation of 33 mixed-discrete engineering optimization problems.[39]

6) Mutation: As stated earlier, mutation is the occasional random alteration on a bit-by-bit basis. Similar to the crossover, there are three mutation operators.

Single-point mutation:

$$1100\underline{1}001 \Rightarrow 1100\underline{0}001$$

Two-point mutation:

$$110\,|\,010\,|\,01 \Rightarrow 110\,|\,101\,|\,01$$

Multipoint mutation:

$$11\underline{00}10\underline{01} \Rightarrow 11\underline{11}10\underline{10}$$

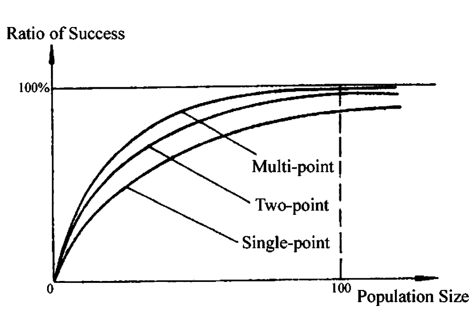

Fig. 1 Comparison of the three crossover strategies.

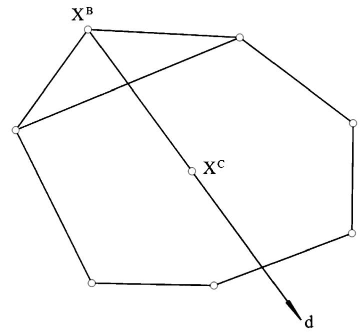

Fig. 2 Shrink direction.

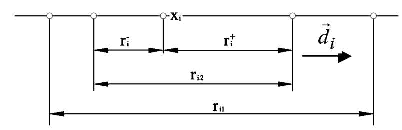

Fig. 3 Shrinking of feasible search region.

Often, the multipoint random mutation proves to be the best choice in this step.

7) Population regeneration: Combine parent population and their offspring into a whole population, sort them in the order from the highest fitness value to the lowest, and select the first M individuals to obtain a new population. The new population is then used to repeat steps 4–7 until the population generated tends to have a stable fitness value; and then go to step 8.

8) Determination of the shrink direction: Once a stable population is obtained, a composite algorithm is applied to find a search direction along which the optimum point can be approached and the feasible region can be shrunken. The composite has some different expressions.[40,41] Here assuming the population as vertices of a composite polygon and vertex X_w as the worst point with the lowest fitness value, vertex X_c that corresponds to the centroid of all vertices except X_w can be defined as

$$X_c = \frac{1}{k-1}\sum_{i=1}^{k} X_i, \qquad X_i \neq X_w \tag{12}$$

A shrink direction for feasible region shown as Fig. 2 can then be determined as

$$\boldsymbol{d} = \frac{X_c - X_w}{\|X_c - X_w\|} \tag{13}$$

This equation takes all individuals into account, better reflects the ecological environment (the distribution and development trend) of the whole population, and hence can work as a favorable search direction.

9) Shrinking of feasible search region: The best individual is taken as the center of the search region, and the feasible region is reduced along the favorable search direction $\boldsymbol{d}$, ensuring that the optimum point lies within the reduced region.

Set r_{i1} as the range of the ith variable before the region is shrunk, r_{i2} as the range after the region is shrunk, r_i^+ as the range along the $+\boldsymbol{d}_i$ direction, and r_i^- as the range along the $-\boldsymbol{d}_i$ direction, and as shown in Fig. 3, we have

$$r_{i2} = r_i^+ + r_i^- \tag{14}$$

$$\gamma_i = r_{i2}/r_{i1} \tag{15}$$

$$\beta_i = r_i^+/r_i^- \tag{16}$$

r_i^+ and r_i^- can be expressed as

$$r_i^+ = \frac{(\gamma_i \beta_i r_{i1})}{(\beta_i + 1)} \tag{17}$$

$$r_i^- = \frac{(\gamma_i r_{i1})}{(\beta_i + 1)} \tag{18}$$

Equations (14–18) indicate that if γ_i and β_i are known, r_i^+ and r_i^- can be derived, and also the range r_{i2} will be known. In this research γ_i is determined according to the following equations:

$$d_{\min} = \min\{|d_i|, i = 1{\sim}n\} \tag{19}$$

$$\gamma_i = \min\{1.0, RA|d_i|/d_{\min}\}, \qquad i = 1{\sim}n \tag{20}$$

where d_i is the ith component of the vector $\boldsymbol{d}$ and RA is the value of γ corresponding to $d_{\min}$. The numerical test of 33 examples found that when β_i changes between 1 and ~3 and RA changes between 70 and ~95% they will have an influence only on the search speed of the algorithm, but not on the optimal results. It was also found that the value of RA can be decreased for problems with fewer design variables and/or constraints and increased for those with more design variables and/or constraints. In the developed algorithm, $\beta_i = 2$ is fixed, and RA is chosen as 85%.

After the new shrunk region is determined, the search goes back to step 2 to repeat the iterative procedure.

10) Finding the optimal solution: When the search region shrinks to a small region within an acceptable precision, the population fitness value will become stable. The best individual of the population is chosen as the starting point for the deterministic discrete iterative method that is used to replace GA to find the final optimum result. The hybrid negative subgradient method combined with discrete one-dimension search is used for this search procedure, and the details can be found in Ref. 39.

3. Analysis of MDHGA

Compared with other GAs used in mixed discrete optimization, the MDHGA not only possesses all features of the GAs but has some distinguishable characteristics and advantages that can be summarized as follows:

1) The MDHGA combines the advantages of random search and deterministic search methods. The GA is used mainly to determine the optimal feasible region surrounding the global optimum point; and the hybrid negative subgradient method integrated with discrete one-dimension search is subsequently used to replace the GA to find the final optimum solution because the deterministic iterative method usually has faster convergence speed and higher computational efficiency compared to the random search method when searching within a fixed small discrete space.

2) The MDHGA uses multipoint crossover and mutation strategy to generate the offspring population that can obtain better fitness values than the regular GAs and speed up the search procedure. This strategy has been proven to have the best ratio of success from numerical experiments.

3) The MDHGA can reduce the computational expense by gradually shrinking the feasible region along the favorable search direction, leading to an algorithm with higher effectiveness and efficiency compared to the regular GAs.

4) The MDHGA can ensure from the following aspects that the global optimal point lies within the gradually reduced search regions to a maximum extent:

a) Consider the worst condition in which the population is evenly distributed throughout the reduced region all of the time during the whole search procedure. Assume the value of the minimum shrink ratio RA as 85%. It means that the probability of the optimal point lying within the shrunk region is at least 85%; and often this probability can guarantee the presence of the global optimum solution in the reduced search region.

b) Before beginning shrinking the search region, the operations of crossover and mutation have already been repeated many times, resulting in the stable trend of the population to the optimal solution. The determination of the shrinking direction also indicates the approximate distribution location of the optimal solution. Reducing the search region at this time can avoid that the next GA operations still work within those regions not covering the optimal point, and hence enhances the computational efficiency.

c) Even if the optimal point is not within the current reduced region, the center point of the search (also the best individual in the population) is continually adjusted to approach the optimal point, which, again, causes the optimal point to be included into the new reduced regions.

5) The algorithm can provide a better initial feasible point for other well-established methods so that their reliability and efficiency can be further improved.

4. Performance Study

An experimental study on MDHGA method can provide an overall evaluation of the performance of the algorithm, indicate the most appropriate fields of application, and provide valuable feedback information to further improve the structure and efficiency of the algorithm.

A total of 33 problems are collected for the present numerical experimental study, which are selected from various past studies on mixed-discrete nonlinear programming reported by Eason and Fenson, Himmmelblau, Gupta and Cha, as well as other literature sources. The detailed information of each test problem, including the objective and constraint functions, the upper and lower bounds of design variables, starting point, optimal solution and computational cost, are given in Ref. 39.

The experimental study has demonstrated that the MDHGA method, combining the advantages of random search and deterministic search methods, can improve the convergence speed and the computational efficiency compared with traditional GAs or some other mixed-discrete methods. It also demonstrated the versatility and robustness of the proposed method to various kinds of mixed-discrete engineering optimization problems, even for large dimensional, highly nonlinear, nondifferentiable and/or nonconvex problems. Several numerical examples given in details in this paper further indicate the reliability and efficiency of the MDHGA when it is used in the MDFMP method for solving mixed-discrete fuzzy multiobjective optimization problems.

C. Computational Procedure

A computer program has been developed to solve multiobjective optimization problems in fuzzy design domains with mixed-discrete design variables. The general computational procedure of MDFMP is summarized in the following steps:

1) Starting from any trial design vector X_0, minimize each of the individual objective functions $f_i(X)$ twice, once with the constraints $g_j(X) \le b_j$ and a second time with the constraints $g_j(X) = b_j + d_j$, $j = 1, 2, \ldots, m$, using the mixed-discrete hybrid genetic algorithm just described. Let the corresponding optimum solutions be denoted as $X^*_{i,c}$ and $X^*_{i,f}$, respectively, $i = 1, 2, \ldots, k$.

2) Construct the matrices $[P_c]$ and $[P_f]$ as

$$[P_c] = \begin{bmatrix} F_1\left(X^*_{1,c}\right) & F_2\left(X^*_{1,c}\right) & \ldots & F_k\left(X^*_{1,c}\right) \\ F_1\left(X^*_{2,c}\right) & F_2\left(X^*_{2,c}\right) & \ldots & F_k\left(X^*_{2,c}\right) \\ \vdots & \vdots & \vdots & \vdots \\ F_1\left(X^*_{k,c}\right) & F_2\left(X^*_{k,c}\right) & \ldots & F_k\left(X^*_{k,c}\right) \end{bmatrix} \tag{21}$$

$$[P_f] = \begin{bmatrix} F_1\left(X^*_{1,f}\right) & F_2\left(X^*_{1,f}\right) & \ldots & F_k\left(X^*_{1,f}\right) \\ F_1\left(X^*_{2,f}\right) & F_2\left(X^*_{2,f}\right) & \ldots & F_k\left(X^*_{2,f}\right) \\ \vdots & \vdots & \vdots & \vdots \\ F_1\left(X^*_{k,f}\right) & F_2\left(X^*_{k,f}\right) & \ldots & F_k\left(X^*_{k,f}\right) \end{bmatrix} \tag{22}$$

3) Define the minimum and maximum possible values of the objective functions as

$$\begin{cases} f_i^{\min} = \min\limits_{j=1}^{k}\left\{f_i\left(X^*_{j,c}\right), f_i\left(X^*_{j,f}\right)\right\}, \\ f_i^{\max} = \max\limits_{j=1}^{k}\left\{f_i\left(X^*_{j,c}\right), f_i\left(X^*_{j,f}\right)\right\}, \end{cases} \qquad i = 1, 2, \ldots, k \tag{23}$$

4) Construct the membership functions corresponding to the fuzzy objectives as

$$\mu_{fi}(X) = \begin{cases} 0, & \text{if} \quad f_i(X) > f_i^{\max} \\ \dfrac{f_i^{\max} - f_i(X)}{f_i^{\max} - f_i^{\min}}, & \text{if} \quad f_i^{\min} < f_i(X) \le f_i^{\max}, \quad i = 1, 2, \ldots, k \\ 1, & \text{if} \quad f_i(X) \le f_i^{\min} \end{cases} \tag{24}$$

5) Represent the fuzzy contraint membership functions using linear relationships (nonlinear relationships can also be used) as

$$\mu_{gj}(X) = \begin{cases} 0, & \text{if} \quad g_j(X) > b_j + d_j \\ 1 - \left(\dfrac{g_j(X) - b_j}{d_j}\right), & \text{if} \quad b_j < g_j(X) \le b_j + d_j, \quad j = 1, 2, \ldots, m \\ 1, & \text{if} \quad g_j(X) \le b_j \end{cases} \tag{25}$$

6) Use the fuzzy λ-formulation technique to solve the mixed-discrete fuzzy multiobjective optimization problem:

Minimize

$$-\lambda$$

Subject to

$$\lambda \le \mu_{fi}(X), \qquad i = 1, \ldots, k$$

$$\lambda \le \mu_{gj}(X), \qquad j = 1, \ldots, m \tag{26}$$

This problem can be solved using the MDHGA programming approaches to find the best compromise solution.

D. Program Flowchart

Figure 4 shows a generalized flowchart of the MDFMP program. The MDFMP program contains the crisp and fuzzy mathematical modeling files, input data file, and MDHGA algorithm module, which is used repeatedly before finding the final optimal solution. In this sense, it is the reliability and robustness of the MDHGA algorithms that decide the precision of the optimum results.

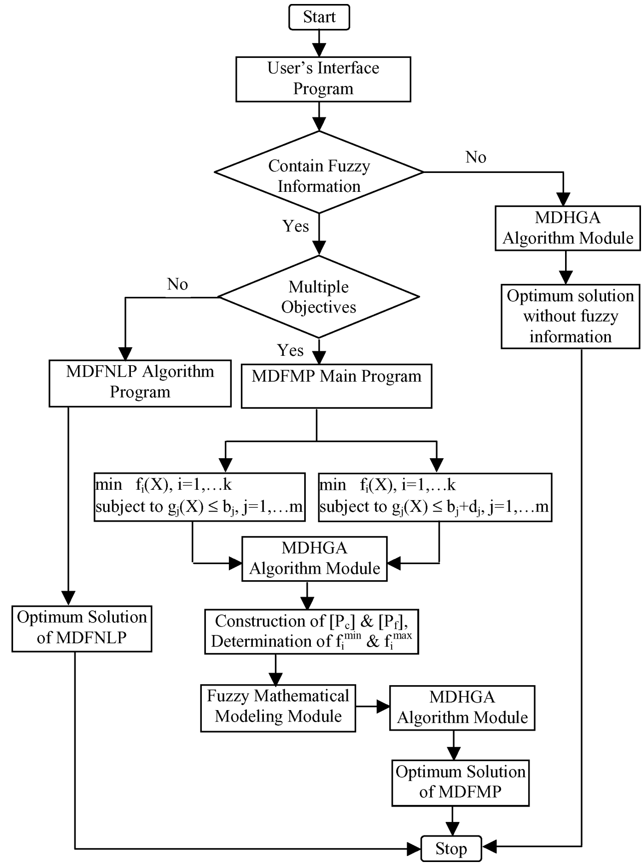

Fig. 4 Flow diagram of the MDFMP program.

III. Illustrative Examples

Three engineering design applications are considered using the MDFMP program to illustrate the reliability and efficiency of the proposed approach. In all of the examples, the fuzzy numerical results are analyzed and compared with the corresponding crisp results.

A. Two-Bar Truss Optimization

One of the most extensively used examples in the optimization literature is the two-bar truss structure shown in Fig. 5. The design of this truss is considered as a simple mixed-discrete fuzzy optimization problem with two objectives. The area of cross section of the bars (A) and the position of the joints 1 and 2 (x) are treated as design variables. The truss is assumed to be symmetric about the y axis. The coordinates of joint 3 are held constant. The weight of the truss and the displacement of joint 3 are considered as the objective functions f_1 and f_2. The stresses induced in the members are constrained to be smaller than the permissible stress σ_0. In addition, lower and upper bounds are placed on the design variables. Thus the problem can be mathematically stated as follows:

Minimize

$$f_1(X) = 2\rho h x_2 \sqrt{1 + x_1^2}$$

$$f_2(X) = \frac{Ph\left(1 + x_1^2\right)^{1.5}\left(1 + x_1^4\right)^{0.5}}{2\sqrt{2}Ex_1^2 x_2} \tag{27}$$

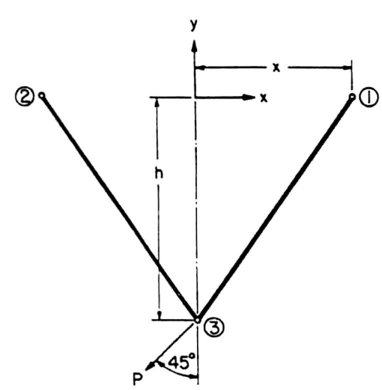

Fig. 5 Two-bar truss.

Subject to

$$g_1(X) = \frac{P(1 + x_1)\left(1 + x_1^2\right)^{0.5}}{2\sqrt{2}x_1 x_2} - \sigma_0 \le 0$$

$$g_2(X) = \frac{P(x_1 - 1)\left(1 + x_1^2\right)^{0.5}}{2\sqrt{2}x_1 x_2} - \sigma_0 \le 0$$

$$g_3(X) = x_1^{(l)} - x_1 \le 0, \qquad g_4(X) = x_2^{(l)} - x_2 \le 0$$

$$0.1 \le x_1 \le 2.5; \qquad x_1 = 0.1 + 0.1k, \qquad k = 0, 1, \ldots, 24$$

$$0.1 \le x_2 \le 2.5; \qquad x_2 = 0.1 + 0.1k, \qquad k = 0, 1, \ldots, 24 \tag{28}$$

where $x_1 = x/h$, $x_2 = A/A_{\min}$, $E = 30e6$ lb/in.^{2} (207 GPa) is the Young's modulus, $\rho = 0.283$ lb/in.^{3} (7.8 Mg/m^3) is the density of the material, and $x_i^{(l)} = 0.1$ is the lower bound on x_i. Other data are chosen as $h = 100$ in. (2.54 m), $A_{\min} = 1$ in.^{2} (6.45e−4 m^2), $P =$ 10,000 lb (4.45e4 N), and $\sigma_0 = 20{,}000$ lb/in.^{2} (137.9 MPa).

Table 1 Optimal results of the two-bar truss[a]

Quantity	Starting point	Min f_1	Min f_2	Crisp optimization[42]: Weighting method ($c_1 = c_2 = 0.5$)	Crisp optimization[42]: Game theory approach	Fuzzy optimization, max λ
Continuous solution						
x_1	1.0	0.6743	0.8612	0.7635	0.7681	——
x_2	1.0	0.5295	2.5	1.0540	1.1408	——
$f_1(X)$	80.044	**36.1493**	186.7361	75.0595	81.4137	——
$f_2(X)$	0.0471	0.0943	**0.0182**	0.0442	0.0408	——
Discrete solution						$\lambda = 0.726$
x_1 (discrete)	1.0	0.5	0.9	——	——	0.8
x_2 (discrete)	1.0	0.6	2.5	——	——	1.1
$f_1(X)$	80.044	**37.9684**	190.3688	——	——	79.7317
$f_2(X)$	0.0471	0.1132	**0.0182**	——	——	0.0417
No. of iterations	——	29	24	——	——	152
No. of fun. calls	——	711	553	——	——	5414

[a]Values in boldface denote the best/final values of the objectives.

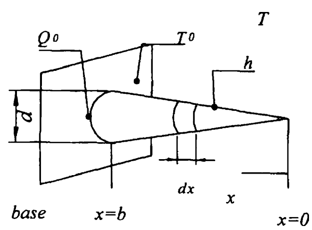

Fig. 6 Conical convective spine.

The constraint tolerances in the fuzzy optimization problem are taken as 10% of their respective stated (original) allowable values. Table 1 shows the discrete optimal solutions obtained in a fuzzy environment using MDFMP program. The ideal discrete solutions for the individual objectives are given by $f_1(X) = 37.968$ and $f_2(X) = 0.018$, close to their respective continuous solutions. The final fuzzy solutions are $f_1(X) = 79.732$ and $f_2(X) = 0.042$ with optimal point $X = (0.8, 1.1)$. The maximum value of $\lambda = 0.726$ indicates that the maximum level of satisfaction (degree of membership) that can be achieved in the presence of the stated fuzziness in the objectives and constraints is 0.726. The computational cost of discrete solution, like number of iterations and number of function calls, is presented in Table 1 for the comparison of single objective and multiobjective optimization. The crisp multiobjective optimization in continuous space[42] is also given for comparison, and in a continuous crisp environment different optimization approaches often bring about different compromise Pareto-optimal solutions. The feasible design space of the problem, along with different design points of interest, is shown graphically in Ref. 42. In a discrete fuzzy environment, a noninferior compromise solution corresponding to the optimal supercriterion can be obtained by the selection of an appropriate mixed-discrete fuzzy programming algorithm. Different multiobjective strategies yield different compromise solutions, and none can be termed more right than the others simply because of the nonunicity of the original design problem statement.

B. Design of a Conical Convective Spine

Figure 6 denotes a conical convective spine.[43] In this study, the problem with two conflicting fuzzy design objectives together with certain constraints is considered using the MDFMP approach. The first objective function, denoted as V, is the spine volume, and the second objective function, denoted as B, is the length of the spine. The problem is to find the length b and diameter d at the base of the spine to minimize both the objectives V and B, subject to the heat-transfer constraint $Q \geq (Q_0 - \nabla Q_0)$, where Q is the actual heat transfer, Q_0 is the required heat transfer, and ∇Q_0 is the allowable variance of Q_0. The variable b is discretized into integer multiples of 0.01 m, and d is taken to be integer multiples of 0.001 m. The simplified mathematical model is expressed as follows:

Minimize

$$V(d, b) = \pi d^2 b/12, \qquad B(b, d) = b \tag{29}$$

subject to

$$Q = \left(\pi \big/ 4\sqrt{b}\right) k d^2 M \theta_0 I_2 / I_1 \tag{30}$$

where $M = \sqrt{(4hb/kd)}$, and I_1 and I_2 are modified Bessel functions. The values of zeroth- and first-order modified Bessel functions $I_0(u)$ and $I_1(u)$ can be found from the table of Bessel functions corresponding to any u, which is defined as $u = 2M\sqrt{b}$. The value of $I_2(u)$ can then be obtained from the equation

$$I_2(u) = I_0(u) - (2/u) I_1(u) \tag{31}$$

In this study, the target values d_{opt} and b_{opt} are determined using the same equations used in the conventional analysis[44]:

$$d_{\text{opt}} = 1.0988 \left[Q_0^2 \big/ hk\theta_0^2\right]^{\frac{1}{3}} \tag{32}$$

$$b_{\text{opt}} = 0.7505 \left[Q_0 k \big/ h^2 \theta_0\right]^{\frac{1}{3}} \tag{33}$$

The minimum value of the spine volume is given by

$$V_{\min} = \pi d_{\text{opt}}^2 b_{\text{opt}} / 12 \tag{34}$$

and the maximum acceptable volume is assumed as

$$V_{\max} = 2 V_{\min} \tag{35}$$

The membership function for the objective V is defined as

$$\mu_V = \begin{cases} 0, & V_{\max} < V \\ \dfrac{V_{\max} - V}{V_{\max} - V_{\min}}, & V_{\min} < V \leq V_{\max} \\ 1, & V \leq V_{\min} \end{cases} \tag{36}$$

The ideal spine length $B_{\min}$ is selected as[44]

$$B_{\min} = 2Q_0 / \pi d_{\text{opt}} h \theta_0 \tag{37}$$

and the worst value is set to be the spine length corresponding to that of $V_{\min}$, that is,

$$B_{\max} = b_{\text{opt}} \tag{38}$$

The membership function of B is specified as

$$\mu_B = \begin{cases} 0, & B_{\max} < B \\ \dfrac{B_{\max} - B}{B_{\max} - B_{\min}}, & B_{\min} < B \leq B_{\max} \\ 1, & B \leq B_{\min} \end{cases} \tag{39}$$

Similarly, the membership function of the constraint is chosen as

$$\mu_Q = \begin{cases} 0, & Q < Q_0 - \Delta Q_0 \\ \dfrac{Q - Q_0 + \Delta Q_0}{\Delta Q_0}, & Q_0 - \Delta Q_0 \leq Q < Q_0 \\ 1, & Q \geq Q_0 \end{cases} \tag{40}$$

The fuzzy multiobjective optimization problem is formulated as follows:

Find

$$X = \begin{Bmatrix} d \\ b \end{Bmatrix}$$

which maximizes

$$f(X) = \lambda$$

subject to

$$\lambda \le \mu_V(X), \qquad \lambda \le \mu_B(X), \qquad \lambda \le \mu_Q(X) \tag{41}$$

The parameters and design data of the conical convective spine are listed in Table 2. Substituting these numerical values into appropriate equations, the boundaries of the fuzzy range in the corresponding objective membership functions can be obtained. The optimum spine designs obtained are given in Table 3. In the continuous design space, the optimal heat transfer is 9.991 W, which is within the allowable limit of 10 W, and the aspect ratio of the variables (b/d) is about 16. In the discrete design space, the aspect ratio came out to be around 16, and the optimal heat transfer is 10.116 W, satisfying the constraint $Q > Q_0$, which means $\mu_Q = 1$. The maximum value of λ is $\lambda = \mu_V = 0.9379$. Although the discrete optimum design variables are equal to the appropriate round-off values of those obtained in the continuous case, the maximum values of λ are not equal and are not determined by the same function with the minimum membership value. For the ideal minimum weight spine the aspect ratio is about 23; hence, the current optimal spine is smaller in size and is easier to manufacture.

C. Twenty-Five-Bar Truss Design

The 25-bar truss shown in Fig. 7 is considered with three objective functions: the minimization of the weight, the minimization of the deflections of nodes 1 and 2, and the maximization of the fundamental natural frequency of vibration of the truss. The truss is required to support the two load conditions given in Table 4 and is designed subject to constraints on member stresses as well as Euler buckling.[42]

The member areas are linked into the following groups: A_1; $A_2 = A_3 = A_4 = A_5$; $A_6 = A_7 = A_8 = A_9$; $A_{10} = A_{11}$; $A_{12} = A_{13}$; $A_{14} = A_{15} = A_{16} = A_{17}$; $A_{18} = A_{19} = A_{20} = A_{21}$; and $A_{22} = A_{23} = A_{24} = A_{25}$. Thus a total of eight independent areas are selected as the discrete design variables. The objective functions can be stated as follows:

Minimize

$$f_1(X) = \sum_{i=1}^{25} \rho A_i l_i$$

$$f_2(X) = \left(\delta_{1x}^2 + \delta_{1y}^2 + \delta_{1z}^2\right)^{\frac{1}{2}} + \left(\delta_{2x}^2 + \delta_{2y}^2 + \delta_{2z}^2\right)^{\frac{1}{2}}$$

$$f_3(X) = -\varpi_1 \tag{42}$$

where l_i is the length of member i; $\rho = 0.1$ lb/in.^{3}; δ_{ix}, δ_{iy}, and δ_{iz} denote the x, y, and z components of the deflection of node i $(i = 1, 2)$; and ω_1 is the fundamental natural frequency of vibration. The constraints can be formulated as

$$|\sigma_{ij}(X)| \le s, \qquad i = 1, 2, \ldots, 25, \qquad j = 1, 2$$

$$\sigma_{ij}(X) \ge p_i(X), \qquad i = 1, 2, \ldots, 25, \qquad j = 1, 2$$

$$x_i^{(l)} \le x_i \le x_i^{(u)}, \qquad i = 1, 2, \ldots, 8 \tag{43}$$

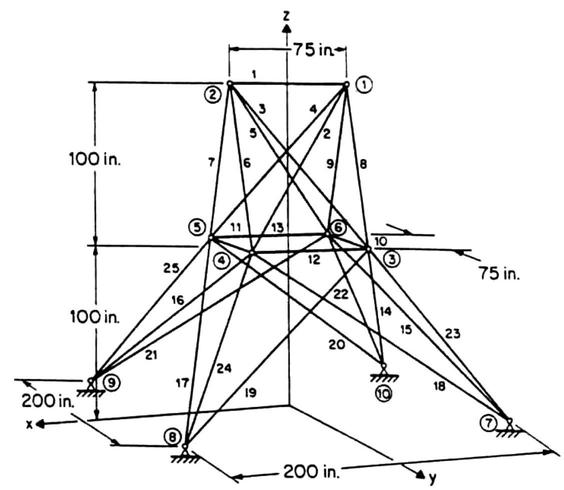

Fig. 7 Twenty-five-bar truss.

Table 4 Loads acting on 25-bar truss

Joint	1, lb	2, lb	3, lb	6, lb
		Load condition 1		
F_x	0	0	0	0
F_y	20,000	−20,000	0	0
F_z	−5,000	−5,000	0	0
		Load condition 2		
F_x	1,000	0	500	500
F_y	10,000	10,000	0	0
F_z	−5000	−5000	0	0

Table 2 Design parameters of the spine

Parameter	Value
Heat conductivity k	200 W/m · °C
Heat-transfer coefficient h	10 W/m^2 · °C
Temperature difference θ_0	100°C
Required heat-transfer rate Q_0	10 W
Allowable variance of Q_0 ΔQ_0	0.2 W

Table 3 Optimal results of the spine

Quantity	Starting point	d_{opt}, b_{opt}	Min V	Min B	Fuzzy optimization, max λ
Continuous case					$\lambda = 0.9626$
d	0.02	0.0188	——	——	0.0217
b	0.4	0.4389	——	——	0.3423
V	4.1888e−5	——	4.0565e−5	8.1129e−5	4.2082e−5
B	0.4	——	0.4389	0.3388	0.3423
Discrete case					$\lambda = 0.9379$
d (discrete)	0.02	0.019	——	——	0.022
b (discrete)	0.4	0.44	——	——	0.34
V	4.189e−5	——	4.1584e−5	8.3168e−5	4.3082e−5
B	0.4	——	0.44	0.3351	0.34
No. of iterations	——	——	——	——	95
No. of fun. calls	——	——	——	——	5002

where σ_{ij} indicates the stress in member i in load condition j; s is the allowable stress, taken as 40,000 psi; p_i denotes the buckling stress in member i given by

$$p_i = \frac{-100.01\pi E A_i}{8l_i^2}, \qquad i = 1, 2, \ldots, 25 \tag{44}$$

E is Young's modulus taken as $E = 10^7$ psi, and $x_i^{(l)}$ and $x_i^{(u)}$ are the lower and upper bounds on x_i, set to be 0.1 in.^{2} and 5.0 in.^{2}, respectively. Each x_i is considered as a discrete variable with permissible values given by the expression

$$x_i = 0.1 + 0.1k, \qquad k = 0, 1, \ldots, 49$$

For the fuzzy problem, the behavior constraints are assumed to have a transition zone defined by

$$d_s = 4000 \text{ psi}, \qquad d_{\text{pi}} = 0.1 \text{ psi}, \qquad d_{x_i^{(l)}} = 0.01 \text{ in.}^2$$

so that the constraints of the fuzzy optimization problem can be stated as

$$|\sigma_{ij}(X)| \le s + d_s \qquad i = 1, 2, \ldots, 25, \qquad j = 1, 2$$

$$\sigma_{ij}(X) \ge p_i(X) - d_{\text{pi}} \qquad i = 1, 2, \ldots, 25, \qquad j = 1, 2$$

$$x \ge x_i^{(l)} - d_{x_i^{(i)}} \qquad i = 1, 2, \ldots, 8 \tag{45}$$

The length of each bar is calculated from the coordinates of the nodes of the truss. The x, y, and z components of the deflection of node i, δ_{ix}, δ_{iy}, and δ_{iz}; the fundamental natural frequency of vibration, ω_1; and the stress in member i in load condition j, σ_{ij}, can be obtained through the finite element analysis of the truss.

The optimization procedure is divided into two stages. In the first stage, the individual objective functions are optimized subject to the constraints of Eqs. (43) and (45) using the MDHGA program. The results are shown in Table 5, together with the results of the corresponding continuous system given in Ref. 42. The best and worst possible values of each of the objective functions, which are used to construct the objective membership functions in the fuzzy formulation, can be identified from these results. It can be seen that $f_1^{\max} = 1605.60$, $f_1^{\min} = 249.32$, $f_2^{\max} = 1.752$, $f_2^{\min} = 0.308$, $f_3^{\max} = -70.226$, and $f_3^{\min} = -113.043$. Table 5 also shows that the ideal maximum frequency in the discrete space is greater than that in the continuous space. This is because the result is obtained in the presence of the fuzzy constraints of Eq. (44), in which the relaxed boundaries are considered.

Table 5 Optimum results of individual objective functions[a]

Quantity	Starting point	Min. of weight	Min. of deflection	Max. of frequency
Continuous results[42]				
x_1	1.0	0.1	3.7931	0.1
x_2	1.0	0.8023	5.0	0.7977
x_3	1.0	0.7479	5.0	0.7461
x_4	1.0	0.1	3.3183	0-.7282
x_5	1.0	0.1245	5.0	0.8484
x_6	1.0	0.5712	5.0	1.9944
x_7	1.0	0.9785	5.0	1.9176
x_8	1.0	0.8025	5.0	4.1119
Weight, lb	330.7208	**233.0727**	1619.3258	600.8789
Deflection, in.	1.5417	1.9250	**0.3083**	1.3550
Frequency, Hz	68.8648	73.2535	70.2082	**108.6224**
Discrete results				
x_1	4.0	0.1	3.0	0.1
x_2	4.0	0.9	5.0	0.8
x_3	4.0	0.9	5.0	0.8
x_4	4.0	0.1	3.0	0.9
x_5	4.0	0.2	4.8	0.1
x_6	4.0	0.6	5.0	4.8
x_7	4.0	1.0	5.0	2.9
x_8	4.0	0.8	5.0	5.0
Weight, lb	1322.8828	**249.3187**	1605.6035	916.5322
Deflection, in.	0.3854	1.7523	**0.3084**	1.2356
Frequency, Hz	68.8884	70.2258	71.1295	**113.0431**

[a]Values in boldface denote the best/final values of the objectives.

In the second stage, the following fuzzy optimization problem is solved: Find $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, \lambda\}$, which minimizes $f(X) = -\lambda$:

subject to

$$\lambda \le \mu_{\text{fi}}(X), \qquad i = 1, 2, 3$$

$$\lambda \le \mu_{\text{gj}}(X), \qquad j = 1, \ldots, 108 \tag{46}$$

where the membership functions are defined by

$$\mu_{f1}(X) = \begin{cases} 0, & \text{if } f_1(X) > 1605.6035 \\ \dfrac{1605.6035 - f_1(X)}{1605.6035 - 249.3187}, & \text{if } 249.3187 < f_1(X) \le 1605.6035 \\ 1, & \text{if } f_1(X) \le 249.3187 \end{cases} \tag{47}$$

$$\mu_{f2}(X) = \begin{cases} 0, & \text{if } f_2(X) > 1.7523 \\ \dfrac{1.7523 - f_2(X)}{1.7523 - 0.3084}, & \text{if } 0.3084 < f_2(X) \le 1.7523 \\ 1, & \text{if } f_2(X) \le 0.3084 \end{cases} \tag{48}$$

$$\mu_{f3}(X) = \begin{cases} 0, & \text{if } f_3(X) > -70.2258 \\ \dfrac{-70.2258 - f_3(X)}{-70.2258 - (-113.0431)}, & \text{if } -113.0431 < f_3(X) \le -70.2258 \\ 1, & \text{if } f_3(X) \le -113.0431 \end{cases} \tag{49}$$

$$\mu_{|\sigma_{ij}|} = \begin{cases} 0, & \text{if } |\sigma_{ij}| > s + b_s \\ 1 - \left(\dfrac{|\sigma_{ij}| - s}{b_s}\right), & \text{if } s < |\sigma_{ij}| \le s + b_s \\ 1, & \text{if } |\sigma_{ij}| \le s \end{cases} \tag{50}$$

$$\mu_{\sigma_{ij}} = \begin{cases} 0, & \text{if } \sigma_{ij} < p_i + d_{p_i} \\ 1 - \left(\dfrac{\sigma_{ij} - p_i}{d_{p_i}}\right), & \text{if } p_i + d_{p_i} \le \sigma_{ij} < p_i \\ 1, & \text{if } \sigma_{ij} \ge p_i \end{cases} \tag{51}$$

$$\mu_{x_i} = \begin{cases} 0, & \text{if } x_i < x_i^l - d_{x_i^l} \\ 1 - \left(\dfrac{x_i^l - x_i}{d_{x_i^l}}\right), & \text{if } x_i^l - d_{x_i^l} \le x_i < x_i^l \\ 1, & \text{if } x_i \ge x_i^l \end{cases} \tag{52}$$

This fuzzy problem is solved to find the optimum results shown in Table 6. The best compromise solution has a weight of 761.0665 lb, a deflection of 0.8311 in., and a fundamental natural frequency of 99.45 Hz, with the maximum level of satisfaction $\lambda = 0.6227$. Because only discrete values can be selected for the variables, no constraints are active at the optimum point. When assuming all design variables are continuous, the same computational procedure is used in the continuous space and shows a different optimal result with a little higher level of satisfaction. It is seen that the optimum values of the design variables x_4, x_5, and x_6 in discrete design space are greater than those in continuous design space, resulting in worse results for the minimum weight and deflection, and a better result

Table 6 Fuzzy optimization results of the 25-bar truss

Quantity	Crisp optimization: game theory approach	Fuzzy optimization: maximization of λ
Continuous	$c_1 = 0.1433$, $c_2 = 0.3628$, $c_3 = 0.4939$	$\lambda_{max} = 0.6918$
x_1	0.1	0.0969
x_2	1.1464	1.3962
x_3	1.3156	1.5619
x_4	0.4838	0.3463
x_5	0.1	0.0969
x_6	1.3315	1.7657
x_7	1.8558	2.1434
x_8	4.4960	4.2812
Weight, lb	596.5181	658.7802
Deflection, in.	0.9401	0.8061
Frequency, Hz	100.2154	96.7849
No. of iterations	——	116
No. of fun. calls	——	5237
Discrete		$\lambda_{max} = 0.6227$
x_1	——	0.1
x_2	——	1.3
x_3	——	1.5
x_4	——	2.8
x_5	——	0.5
x_6	——	2.6
x_7	——	2.0
x_8	——	4.6
Weight, lb	——	761.0665
Deflection, in.	——	0.8311
Frequency, Hz	——	99.4500
No. of iterations	——	104
No. of fun. calls	——	5154

for the maximum frequency. The conflicting nature of the objectives still exists in a fuzzy environment. The comparison of the computational costs for discrete solution and continuous solution in a fuzzy environment indicates that when the MDFMP method is used for continuous optimization problems the fast convergence speed and high computational efficiency can still be achieved. The crisp optimal design obtained in Ref. 42 using the game theory approach in the continuous space is also given in Table 6, but there is no computational cost reported from the literature for the comparison with fuzzy optimization.

IV. Conclusions

A new programming method called mixed-discrete fuzzy multiobjective programming (MDFMP) is proposed in this paper, in which the fuzzy λ formulation, the game theory technique, and a mixed-discrete hybrid genetic algorithm (MDHGA) are combined, for solving mixed-discrete fuzzy multiobjective optimization problems. The fuzzy programming algorithm coupled with game theory can promise a noninferior compromise solution corresponding to the optimal supercriterion. The MDHGA method, combining the advantages of random search and deterministic search methods, can improve the convergence speed and the computational efficiency compared with traditional genetic algorithms or other mixed-discrete methods. The present work represents the first attempt at solving multiobjective programming problems in a mixed-discrete fuzzy environment. Although the computational cost is high for large-scale problems, the MDFMP method can be effectively applied to various kinds of simple or complex engineering optimization problems, even for highly nonlinear, nondifferentiable, and/or nonconvex problems, to obtain more realistic and satisfied results in a fuzzy environment. Three numerical examples—the optimal designs of a two-bar truss, a conical convective spine, and a 25-bar truss—are used to illustrate the practical application of the present method and demonstrate its reliability and efficiency as an advanced global optimum approach.

References

[1]Kuhn, H. W., and Tucker, A. W., "Nonlinear Programming," *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, edited by J. Neyman, Univ. of California, Berkeley, CA, 1951, pp. 481–491.

[2]Zadeh, L. A., "Fuzzy Sets," *Information and Control*, Vol. 8, 1965, pp. 338–353.

[3]Bellman, R. E., and Zadeh, L. A., "Decision Making in a Fuzzy Environment," *Management Science*, Vol. 17, 1970, pp. 141–164.

[4]Tang, J., and Wang, D., "Modeling and Optimization for a Type of Fuzzy Nonlinear Programming Problems in Manufacturing Systems,"*Proceedings of the 35th IEEE Conference on Decision and Control*, 1996, pp. 4401–4405.

[5]Chen, L., and Rao, S. S., "A Fuzzy Approach for the Manipulation of Uncertainties in Determination of Optimum Machining Conditions Under Multiple Criteria," *Journal of Manufacturing Science and Engineering*, Vol. 119, No. 2, 1997, pp. 186–192.

[6]Chen, Y.-D., Hui, A., and Du, R., "Fuzzy Expert System for the Optimal Design of Machining Operations," *International Journal of Machine Tools and Manufacture*, Vol. 35, No. 12, 1995, pp. 1605–1621.

[7]Abdelaziz, A. R., and Gouda, S. A. M., "Geno-Fuzzy Optimization of Power System Reliability," *Proceedings of the Mediterranean Electrotechnical Conference—MELECON, 10th Mediterranean Electrotechnical Conference*, 2000, pp. 991–994.

[8]Guan, X.-H., Liu, W.-H. E., and Papalexopoulos, A. D., "Application of a Fuzzy Set Method in an Optimal Power Flow," *Electric Power Systems Research*, Vol. 34, 1995, pp. 11–18.

[9]Xiang, H., Verma, B. P., and Hoogenboom, G., "Fuzzy Irrigation Decision Support System," *Proceedings of the 12th National Conference on Arificial Intelligence: Part 2*, 1994.

[10]Teegavarapu, R. S. V., and Simonovic, S. P., "Modeling Uncertainty in Reservoir Loss Functions Using Fuzzy Sets," *Water Resources Research*, Vol. 35, No. 9, 1999, pp. 2815–2823.

[11]Sheng, W., Miao, D., and Dai, R., "Two-Level Optimization Strategy for Fuzzy Control Design," *Proceedings of the 1997 IEEE International Conference on Intelligent Processing Systems, ICIPS'97: Part 1*, 1998, pp. 276–280.

[12]Sutton, R., "Optimization of Fuzzy Autopilots," *Proceedings of the 1997 11th Ship Control Systems Symposium: Part 1*, 1997, pp. 63–76.

[13]Zhao, D., and Zhu, J., "Fuzzy Variable-Structure-Adaptive Control Design for Robot Manipulators," *CIRP Annals—Manufacturing Technology*, Vol. 44, No. 1, 1995, pp. 363–368.

[14]Buckley, J. J., "Fuzzy Programming and the Pareto Optimal Set," *Fuzzy Sets and Systems*, Vol. 10, No. 1, 1983, pp. 57–63.

[15]Kassem, M. A. E.-H., "Study of Fuzzy Parametric on Multiobjective Nonlinear Programming Problems Without Differentiability," *Journal of Engineering and Applied Science*, Vol. 98, No. 1, 1998, pp. 67–75.

[16]Osman, M. S., and El-Banna, A.-Z. H., "Stability of Multiobjective Nonlinear Programming Problems with Fuzzy Parameters," *Mathematics and Computers in Simulation*, Vol. 35, No. 4, 1993, pp. 321–326.

[17]Ammar, E. E., "Stability of Multiobjective NLP Problems with Fuzzy Parameters in the Objectives and Constraints Functions," *Fuzzy Sets and Systems*, Vol. 90, No. 3, 1997, pp. 225–234.

[18]Mohan, C., and Nguyen, H. T., "Interactive Satisficing Method for Solving Multiobjective Mixed Fuzzy-Stochastic Programming Problems," *Fuzzy Sets and Systems*, Vol. 117, No. 1, 2001, pp. 61–79.

[19]Fadale, T., and Sues, R. H., "Reliability-Based Analysis and Optimal Design of an Integral Airframe Structure Lap Joint," AIAA Paper 99-1604, 1999.

[20]Rao, S. S., and Das, G., "Reliability-Based Optimum Design of Gear Trains," *Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 106, 1984, pp. 17–22.

[21]Tappeta, R. V., Renaud, J. E., Messac, A., and Sundararaj, G. J., "Interactive Physical Programming: Tradeoff Analysis and Decision Making in Multidisciplinary Optimization," *AIAA Journal*, Vol. 30, No. 5, 2000, pp. 917–926.

[22]Messac, A., "Physical Programming: Effective Optimization for Computational Design," *AIAA Journal*, Vol. 34, No. 1, 1996, pp. 149–158.

[23]Chen, L., and Rao, S. S., "A Modified Dempster-Shafer Theory for Multicriteria Optimization," *Engineering Optimization*, Vol. 30, 1998, pp. 171–201.

[24]Rao, S. S., and Dhingra, A. K., "Reliability and Redundancy Apportionment Using Crisp and Fuzzy Multiobjective Optimization Approaches," *Reliability Engineering & System Safety*, Vol. 37, No. 3, 1992, pp. 253–261.

[25]Dhingra, A. K., and Rao, S. S., "A Cooperative Fuzzy Game Theoretic Approach to Multiple Objective Design Optimization," *European Journal of Operations Research*, Vol. 83, 1995, pp. 547–567.

[26]Rao, S. S., Sundararaju, K., Prakash, B. G., and Balakrishna, C., "Fuzzy Goal Programming Approach for Structural Optimization," *AIAA Journal*, Vol. 30, No. 5, 1992, pp. 1425–1432.

[27]Gen, M., Ida, K., and Kim, J., "Fuzzy Goal Programming Using Genetic Algorithm," *Proceedings of the 1997 IEEE International Conference on Evolutionary Computation*, ICEC'97, 1997, pp. 413–418.

[28]Sakawa, M., Yano, H., and Sawada, K., "Fuzzy Dual Decomposition Method for Large-Scale Multiobjective Nonlinear Programming Problems with Block Angular Structure," *Electronics & Communications in Japan, Part III: Fundamental Electronic Science*, Vol. 78, No. 6, 1995, pp. 86–94.

[29]Sakawa, M., Yano, H., and Sawada, K., "Primal Decomposition Method for Multiobjective Structured Nonlinear Programs with Fuzzy Goals," *Cybernetics and Systems*, Vol. 26, No. 4, 1995, pp. 413–426.

[30]Sakawa, M., Yumine, T., and Yano, H., *Interactive Fuzzy Satisficing Method for Multiobjective Nonlinear Programming Problems*, CRC Press, Inc., Boca Raton, FL, 1987, pp. 273–283.

[31]Sakawa, M., and Yano, H., "Interactive Fuzzy Decision Making for Multiobjective Nonlinear Programming Using Augmented Mimimax," *Fuzzy Sets and Systems*, Vol. 20, No. 1, 1986, pp. 31–43.

[32]Sakawa, M., Yumine, T., and Nango, Y., "Interactive Fuzzy Decision Making for Multiobjective Nonlinear Programming Problems," *Electronics and Communications in Japan*, Vol. 67, No. 4, 1984, pp. 31–38.

[33]Sakawa, M., and Yano, H., "Interactive Fuzzy Satisficing Mentod for Multiobjective Nonlinear Programming Problems with Fuzzy Parameters," *Electronics and Communications in Japan, Part 1: Communications*, Vol. 70, No. 2, 1987, pp. 25–34.

[34]Sakawa, M., and Yauchi, K., "Interactive Fuzzy Satisficing Method for Multiobjective Nonconvex Programming Problems Through Floating-Point Genetic Algorithms," *Electronics and Communications in Japan, Part III: Fundamental Electronic Science*, Vol. 83, No. 6, 2000, pp. 10–18.

[35]Rao, S. S., "Description and Optimum Design of Fuzzy Mechanical Systems," *Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 109, 1987, pp. 126–132.

[36]Xiong, Y., and Rao, S. S., "Fuzzy Nonlinear Programming for Mixed-Discrete Design Optimization Through Hybrid Genetic Algorithm," *Fuzzy Sets and Systems*, Vol. 146, No. 2, 2004, pp. 167–186.

[37]Holland, J. H., *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press, Ann Arbor, MI, 1975.

[38]Rudolph, G., "Convergence Analysis of Canonical Genetic Algorithms," *IEEE Transactions on Neural Network*, Vol. 5, No. 1, 1994, pp. 96–101.

[39]Xiong, Y., "Mixed Discrete Fuzzy Nonlinear Programming for Engineering Design Optimization," Ph.D. Dissertation, Dept. of Mechanical Engineering, Univ. of Miami, Coral Gables, FL, 2002.

[40]Fox, D. B., "A Composite Algorithm for Mixed Integer Constrained Nonlinear Optimization," *Engineering Optimization*, Vol. 5, 1981, pp. 129–149.

[41]Chen, L. Z., "Mixed Discrete Composite Algorithm and Programming," *Journal of Beijing Institute of Science and Technology*, Vol. 9, No. 4, 1987.

[42]Rao, S. S., "Game Theory Approach for Multiobjective Structural Optimization," *Computers and Structures*, Vol. 25, No. 1, 1987, pp. 119–127.

[43]Ken, D. Q., and Kraus, A. D., *Extended Surface Heat Transfer*, McGraw–Hill, New York, 1972.

[44]Chen, C.-K., "A Multi-Objective Fuzzy Optimization for Optimum Dimensions Design of a Convective Spine," *International Communication in Heat Mass Transfer*, Vol. 28, No. 1, 2001, pp. 67–76.

A. Messac
Associate Editor