http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (23): 7646-7650, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.7646.7650

Multi-objective Structural Optimization Base on Improved NSGA-II Algorithm

[1]M.Z. Lai, [2]Z.M. Duan, [3]G.Y Zhang and [3]B.D
[1]College of Computer Science and Technology, Harbin University of Science and Technology, Harbin, Heilongjiang 150080, China
[2]HLJ Province Electronic and Information Products Supervision Inspection Institute, Harbin, Heilongjiang, 150090, China
[3]School of Astronautics, Harbin Institute of Technology, Heilongjiang150001, China

Abstract: A kind of fast and elitist multi-objective genetic algorithm (nondominated sorting genetic algorithm -II) was presented to solve high dimension and multi-modal optimal problems. T His fuzzy information could be converted into a mathematically well-structured problem based on fuzzy optimal theory. And the improved crossover operator of NSGA-II was applied to obtain the optimal solution. According to the test results on a typical test function and an application on the structural fuzzy multi-objective optimization of three-bar truss, more reasonable distributed solutions could be obtained and the diversity of the solutions could be maintained. It provides beneficial references for engineering application of fuzzy multi-objective structure optimization.

Key words: Structural optimization, fuzzy multi-objective, nondominated sorting genetic algorithm-II (NSGA-II)

INTRODUCTION

In the initial phase of engineering structures design, there contains a lot of uncertain information, such as random information and fuzzy information and so on and the optimal design of engineering structures is normally reflected in a multi-objective form. For the traditional multi-objective optimization problem, the most direct solution is to use weighted sum method to integrate the various objective functions into one single-objective optimization problem. Conceptually, the weighted sum method could be seen as multi-objective optimization methods used in the genetic algorithm. The method will assign different weight factors to each of the objective functions and then combine the weighted object to a single objective function. In fact, weighted sum methods used in the genetic algorithm are essentially different from weighted sum method used in the traditional multi-objective optimization algorithm (Fonseca and Fleming, 1995; Rubenstein-Montano *et al*., 2000). In multi-objective optimization, weighted sum methods are used to obtain a compromise solution. The only operation requirement of this algorithm is the appropriate weight vector. But for a given problem it's often difficult to obtain a group of suitable weight factors. In the genetic algorithm, the initial weighted sum methods are used for genetic search along to Pareto. With the process of the evolution, the adaptive weight factors need to readjust. So it does not necessarily require a good weight vector to ensure the genetic algorithm operation. In addition, the weights' Shortcomings in the traditional multi-objective optimization could also be weakened based on the population search and revolutionary search. The research of multi-objective genetic algorithm is the mechanism using Pareto multi-objective optimization technology (Zitzler and Thiele, 1999), therefore Nondominated Sorting Genetic Algorithm-II (NSGA-II) could be used to solve the multi-objective fuzzy optimization problem.

CACULATION PROGRESS OF FUZZY MULTI-OBJECTIVE OPTIMATION BASED ON NSGA-II

The Nondominated Sorting Genetic Algorithm (NSGA) was introduced by (Srinivas and Deb, 1994). The algorithm is realized by individual classification, so NSGA is widely used in solving practical problems. However it is not adequate for high-dimensional and multi-modal issues. In 2000, Deb NSGA improved NSGA as NSGA-II (Deb *et al*., 2000), so the computational speed and the robustness of the algorithm were enhanced greatly. In order to demarcate the fitness of small elements in the same level after fast-grade non-winners ranking, so the prospective Pareto jurisdictions elements can be extended to the entire Pareto domain as evenly as possible to fill the

Corresponding Author: M.Z. Lai, College of Computer Science and Technology, Harbin University of Science and Technology, Harbin, Heilongjiang 150080, China

whole space, the concept of crowding distance was put forward, using more crowded distance operator to replace the need to share complex calculation parameters of appropriate sharing methods.

Self-crossover operator design of NSGA-II: Generally, real-coded genetic algorithm is adopted, cross operator adopts arithmetic crossover:

$$c_{1,k} = \frac{1}{2}[(1-\beta_k)p_{1,k} + (1+\beta_k)p_{2,k}] \quad (1)$$

$$c_{2,k} = \frac{1}{2}[(1+\beta_k)p_{1,k} + (1-\beta_k)p_{2,k}] \quad (2)$$

where, $C_{i,k}$ is the kth individual of the ith generation; $p_{1,k}$ and $p_{2,k}$ are the choice of the parent generation; β_k represents a random number of population density.

Now by improving the random number, we can combine this random number with the ranking grades of each individual in the populations, namely:

$$\beta_k = \frac{\text{rankA} * \text{rankB}}{\text{rank}^2\text{A} + \text{rank}^2\text{B}} \quad (3)$$

where, renkA and rankB are, respectively the ordination of A and B populations.

So with the development of the evolution, the individuals from the population will tend to the same surface of Pareto, the sequence of A will be consistent with B and β_k tends to 0.5.

Clearing fuzzy constraint function: Linear membership function was utilized in constraint function to form different transformation forms from the non-Fuzzy to Fuzzy its multi-objective fuzzy optimization model can be converted into the following mathematical model:

$$\begin{aligned} &\text{Find} && X = [x_1, x_2 \cdots x_n]^T \in R^n \\ & && \min f_i(x) \quad i = 1, 2, \cdots n \\ &\text{s.t.} && \mu_{gi} \geq w_{gi}\lambda^* \\ & && \mu^l_{xk} \geq w_{xk}\lambda^* \\ & && \mu^u_{xm} \geq w \end{aligned} \quad (4)$$

where, w_{gi}, w_{xk} and w_{xm} are the weight factors of performance constraints and design variable constraints; μ_{gi} is the memberships of performance constraints; μ^l_{xk} and μ^u_{xm} are the up and down limits of design variable constraints; λ^* is the level cut sets.

The Eq. (4) uses different constraints in different design level. The objective functions use linear weighted method so as to make the objective function and constraint function in different important degrees and meet the actual engineering requirements.

Calculation flow of the improved NSGA-II: Firstly, according to Eq. (4), multi-objective constraint function or design variable constraint is cleared by using linear membership functions, then making the optimization based on the improve NSGA-II. Finally, the global optimal solution is obtained, as shown in Fig. 1.

TEST FUNTIONS AND THE RESULTS

After the linear membership function was utilized in constraint function, the results of test functions are given directly here:

$$\begin{aligned} & c_{1,k} = \frac{1}{2}[(1-\beta_k)p_{1,k} + (1+\beta_k)p_{2,k}] \\ & \text{Find} \quad \min f_i(x) \quad i = 1, 2, 3 \\ & \begin{cases} f_1(x) = (1+g(x))\cos(0.5\pi x_1)\cos(0.5\pi x_2) \\ f_2(x) = (1+g(x))\cos(0.5\pi x_1)\sin(0.5\pi x_2) \\ f_3(x) = (1+g(x))\sin(0.5\pi x_2) \end{cases} \\ & \text{s.t.} \quad g(x) = \sum_1^{12}(x_i - 0.5)^2 \\ & \qquad 0 \leq x_i \leq 1 \quad i = 1, 2, \cdots, 12 \end{aligned} \quad (5)$$

Programmed with MBASIC language, population initialization was shown in Table 1.

When hereditary algebra is 1000 and 2000, Perato space solution sets of NSGA-II were shown in Fig. 2 and 3.

The improved NSGA-II population parameters were shown in Table 2, NSGA-II Perato space solution sets of hereditary algebra 1000 were shown in Fig. 4.

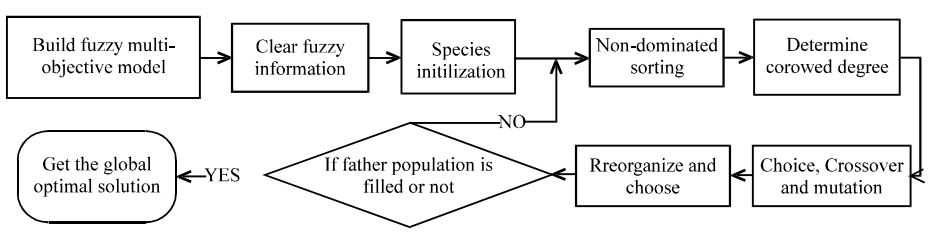

Fig. 1: Calculation flow of the improved NSGA-II

Table 1: Initialized population parameters for NSGA-II

Population size	Binary crossover operator	Mutation operator	Mating pool	Crossover probability	Variation probability
200	20	20	00	0.95	0.05

Table 2: Initialized population parameters with the improved NSGA-II

Population size	Hereditary algebra	Binary crossover operator	Mutation operator	Mating pool	Crossover probability	Variation probability
200	1000	20	20	100	0.95	0.05

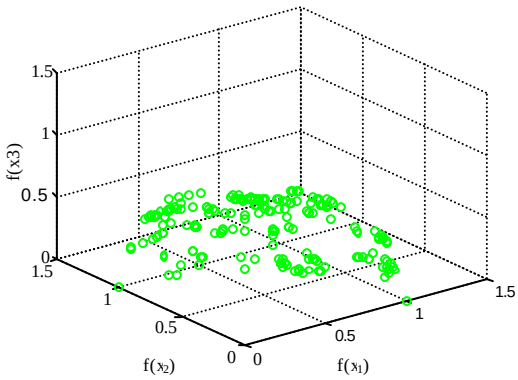

Fig. 2: Perato space solutions with hereditary algebra 1000

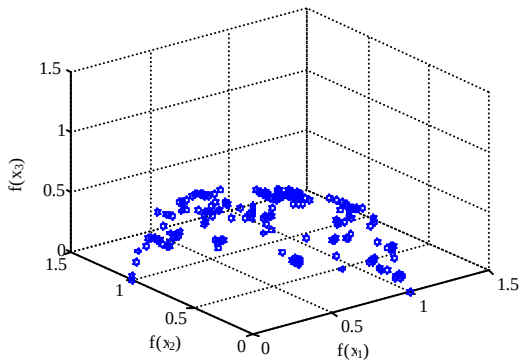

Fig. 3: Perato space solutions with hereditary algebra 2000

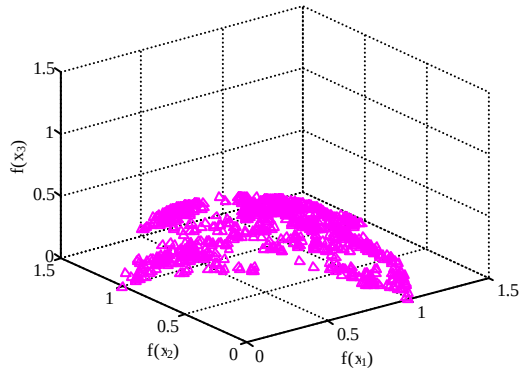

Fig. 4: Perato space solutions of improved NSGA-II with hereditary algebra 1000

APPLICATION OF NSGA-II IN THE FUZZY STRUCTUAL OPTIMATION

A fuzzy optimization design for the three-bar truss' minimum quality was shown in Fig. 5. The known parameters are as follows:

$$\rho = 20,\ H = 1,\ \rho = 1,\ E = 1$$

The allowable limitations and tolerance of each weight were shown in Table 3.

The vertical displacement function $f_1(X)$ of the truss' load location and the weight function $f_2(X)$ are the minimum optimal goals. So it is a multi-objective fuzzy optimization problem.

Set $A_1 = A_3 = x_1$, $A_2 = x_2$. And the truss' fuzzy optimization model is shown as follows.

According to the constraint conditions, we set the stress design standards as $\lambda_1 = \lambda_2 = \lambda_3$, the boundary design standards is λ_4. For calculating simple, setting $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 1$. Solve the question with NSGA-II. Population parameters were shown in Table 4.

The perato solution space of NSGA-II with hereditary algebra 200 and 1000 were shown in Fig. 6 and 7. which reflects that distribution of the Perato space solution with hereditary generation 200 and 1000 is almost the same. So the space solution almost could not be improved after hereditary generation 200 with NSGA-II algorithm.

$$\begin{aligned}
&\text{Find} && X = [x_1, x_2]^T \\
&\min && f_1(X) = 20/(x_1 + \sqrt{2}x_2) \qquad \text{(The objective is the truss's load location)} \\
& && f_2(X) = 2\sqrt{2}x_1 + x_2 \\
&\text{s.t.} && 20\sqrt{2}(x_1 + x_2)/(\sqrt{2}x_1^2 + 2x_1x_2) \le 20 + 4(1-\lambda_1) \qquad \text{(Bar 1's stress}\sigma_1(X)) \\
& && 20\sqrt{2}x_1/(\sqrt{2}x_1^2 + 2x_1x_2) \le 20 + 4(1-\lambda_2) \qquad \text{(Bar 2's stress}\sigma_2(X)) \\
& && -20x_2/(\sqrt{2}x_1^2 + 2x_1x_2) \le -15 - 3(1-\lambda_3) \qquad \text{(Bar 3's stress}\sigma_3(X)) \\
& && 0.1 - 0.02(1-\lambda_4) \le x_i \le 3 + 0.6(1-\lambda_4) \qquad i = 1, 2 \\
& && \lambda_i \in [0,1] \quad i = 1, 2 \cdots, 4
\end{aligned} \tag{6}$$

The improved multi-objective NSGA-II population parameters of fuzzy optimization design for the three-bar truss' minimum quality were shown in Table 5, the Perato solution space of improved NSGA-II with hereditary algebra 200 were shown in Fig. 8. Compare the NBGA-II, the improved NSGA-II distributes

Table 3: Allowable limitations and tolerance of each weight factor

σ^u	σ^u_σ	σ^l	σ^l_σ	A^u	A^u_A	A^l	A^l_A	u^u	d^u_u
20	4	-15	3	3	0.6	0.1	0.02	10/E	2/E

Table 4: Initialized population parameters for fuzzy multi-objective NSGA-II

Population size	Binary crossover operator	Mutation operator	Mating pool	Crossover probability	Variation probability
200	20	20	100	0.95	0.05

Table 5: Initialized population parameters for fuzzy multi-objective improved NSGA-II

Population size	Hereditary algebra	Binary crossover operator	Mutation operator	Mating pool	Crossover probability	Variation probability
200	200	20	20	100	0.95	0.05

Table 6: Numerical results with different methods

Genetic algorithm	Hereditary algebra	x_1	x_2	$f_1(x)$	$f_2(x)$
NSGA-II	200	0.5738	3.1353	3.9938	4.7583
	1000	0.5736	3.1357	3.9935	4.7581
Improved NSGA-II	200	0.5732	3.1358	3.9937	4.7571

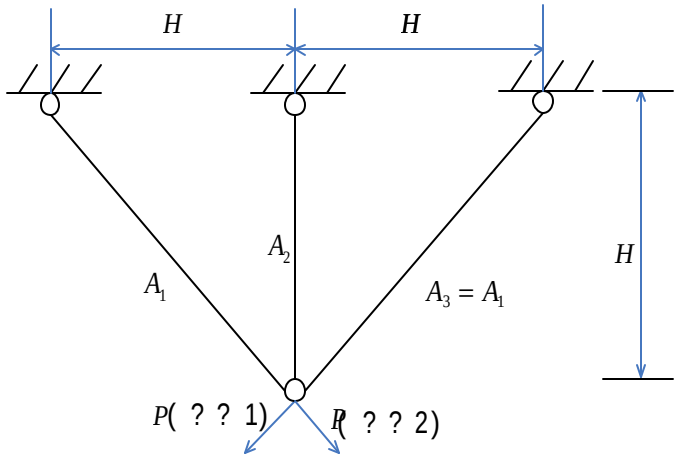

Fig. 5: Three-bar truss

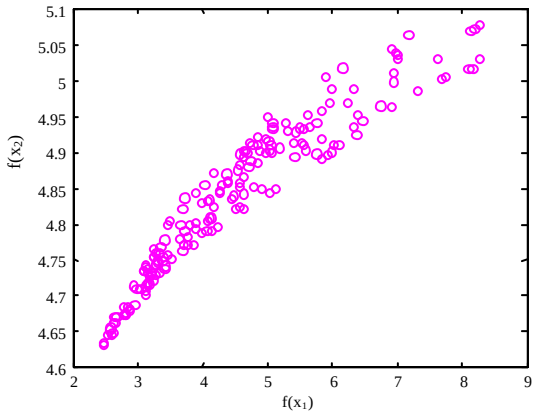

Fig. 6: Perato space solutions with hereditary

more reasonably and easily gets the optimal global solution. The results were shown in the Table 6.

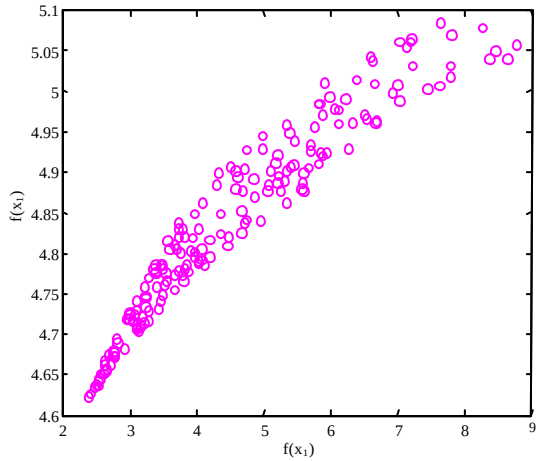

Fig.7: Perato space solutions with hereditary generation 200 generation 1000

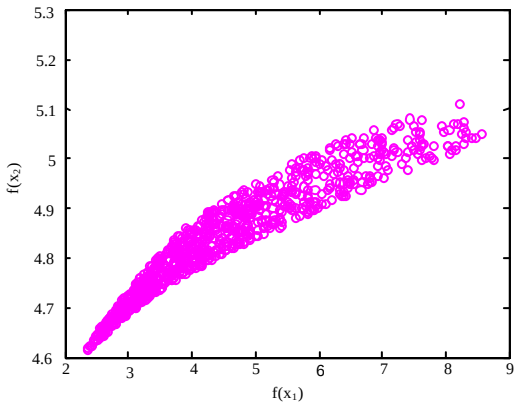

Fig. 8: Perato space solutions of improved NSGA-II with hereditary algebra 200

CONCLUSION

The traditional method to deal with fuzzy multi-objective optimization is more likely to lose the global optimal solution. Furthermore, the choice of the different levels of solution sets and weights needs more engineering experience.

NSGA-II solution is the optimal solution sets from the beginning to the current population. The elitist solution is always believed as a better way to improve the algorithm effiency. The application of improved muti-objective NSGA-II algorithm of this paper indicates that the elitist policy keeps the elitist non-inferior solution from the former generation to current and makes the funtion of the elitist solution continue. By improving cross operator, regulating the parameter and improving the genetic elitist policy to control the sulotion range in the genetic operation, some non-optimal solutions join the genetic operation. From the figures of the final Pareto solution sets, convergence property had changed greatly. There also obtained a better balance between deviersity and convergence.

ACKNOWLEDGMENTS

This research was supported by the Harbin City Key Technologies R and D Program under Grant No. 2011AA1BG059 and Key Program of National Natural Science Foundation of Heilongjiang No. ZD201309.

REFERENCES

Deb, K., S. Agrawal, A. Pratap and T. Meyarivan, 2000. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. Proceedings of the 6th International Conference on Parallel Problem Solving from Nature, September 18-20, 2000, Paris, pp: 849-858.

Fonseca, C.M. and P.J. Fleming, 1995. An overview of evolutionary algorithms in multiobjective optimization. Evol. Comput., 3: 1-16.

Rubenstein-Montano, B. and R.A. Malaga, 2000. A weighted sum genetic algorithm to support multiple-party multiple-objective negotiations. IEEE Trans. Evol. Comput., 6: 366-377.

Srinivas, N. and K. Deb, 1994. Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evol. Comput., 2: 221-248.

Zitzler, E. and L. Thiele, 1999. Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. IEEE Trans. Evolutionary Comput., 3: 257-271.