The Enhanced Genetic Algorithms for the Optimization Design

Pengfei Guo Xuezhi Wang Yingshi Han
School of Civil and Architectural Engineering
Liaoning University of Technology
Jinzhou 121001, China

Abstract **—Three different kinds of the novel enhanced genetic algorithm procedures including the hybrid genetic algorithm, interval genetic algorithm and hybrid interval genetic algorithm are respectively presented. As the results of the proven systems show, the hybrid genetic algorithm can determines the better optimum design than the traditional optimization algorithms and genetic algorithm. The interval genetic algorithm and hybrid interval genetic algorithm can avoid calculating system slope in traditional interval analysis and determines the optimum interval range of the parameters under allowable corresponding objective error boundary. It is the first time that genetic algorithm has been applied to interval optimization process.**

Keywords —optimization; genetic algorithms; hybrid genetic algorithm; interval genetic algorithm; hybrid interval genetic algorithm

I. INTRODUCTION

Optimization is the process of making something better. In engineering, optimization algorithms have been extensively developed and well used in all respects for a long time. An engineer or a scientist conjures up a new idea and optimization improves on that idea. Optimization consists in trying variations on an initial concept and using the information gained to improve on the idea. Many optimization problems from the industrial engineering world, in particular the manufacturing systems, are very complex in nature and quite hard to solve by conventional optimization techniques.

Genetic Algorithm (GA) is one of the optimization algorithms, which is invented to mimic some of the processes observed in natural evolution. The Genetic Algorithm is stochastic search techniques based on the mechanism of natural selection and natural genetics. That is a general one, capable of being applied to an extremely wide range of problems. The GA, differing from conventional search techniques, start with an initial set of random solutions called population. Each individual in the population is called a chromosome, representing a solution to problem at hand. The chromosomes evolve through successive iterations, called generations. During each generation, the chromosomes are evaluated, using some measures of fitness. To create the next generation, new chromosomes, called offspring, are form by either merging two chromosomes form current generation using a crossover operator or modifying a chromosome using a mutation operator. A new generation is form by selecting, according to the fitness values, some of the parents and offspring; and rejecting others so as to keep the population size constant. Fitter chromosomes have higher probabilities of being selected. After several generations, the algorithms converge to the best chromosome, which hopefully represents the optimum or suboptimal solution to the problem.

The GA has received considerable attention regarding their potential as a novel optimization technique. There are three major advantages when applying the GA to optimization problems.

● The GA do not have much mathematical requirements about the optimization problems. Due to their evolutionary nature, the GA will search for solutions without regard to the specific inner workings of the problem.

● The evolution operators make GA effective at performing global search. The traditional approaches perform local search by a convergent stepwise procedure, which compares the values of nearby points and moves to the relative optimal points. Global optimum can be found only if the problem possesses certain convexity properties that essentially guarantee that any local optimum is a global one.

● GA provide a great flexibility to hybridize with domain dependent heuristics to make an efficient implementation for a specific problem.

In the above statement indicate, the GA have much advantages. Even though the GA can locate the solution in the whole domain, it does not solve complexes constraint problems easily, especially for exact constraints. And huge evaluations for generations and populations sometime are time-consuming. To account some of the defects and employ the advantages of the GA, the enhanced GA is proposed and applied for the optimization design.

II. GENETIC ALGORITHM

Genetic Algorithm (GA), first introduced by John Holland in the early seventies, is the powerful stochastic algorithm based on the principles of natural selection and natural genetics, which has been quite successfully, applied in machine learning and optimization problems. To solve a problem, a GA maintains a population of individuals (also called strings or chromosomes) and probabilistically modifies the population by some genetic operators such as selection,

This project is supported the National Science Foundation of China, (No.50709013)

crossover and mutation, with the intent of seeking a near-optimal solution to the problem.

A. Coding to Strings

In GA, each individual in a population is usually coded as coded as a fixed-length binary string. The length of the string depends on the domain of the parameters and the required precision. For example, if the domain of the parameter x is [-2,5] and the precision requirement is six places after the decimal point, then the domain [-2,5] should be divided into 7,000,000 equal size ranges. The implies that the length of the string requires to be 23, for the reason that

$$4194304=2^{22}<7000000<2^{23}=8388608$$

The decoding from a binary string $<b_{22}b_{21}\ldots b_0>$ into a real number is straightforward and is completed in two steps.

(1). Convert the binary string $<b_{22}b_{21}\ldots b_0>$ from the base 10 by

$$x'=\sum_{i=0}^{22} b_i 2^i \tag{1}$$

(2). Calculate the corresponding real number x by

$$x=-2.0+x'\frac{7}{2^{23}-1} \tag{2}$$

B. Initial Population

The initial process is quite simple. We create a population of individuals, where individual in a population is a binary string with a fixed-length, and every bit of the binary string is initialized randomly.

C. Evaluation

In each generation for which the GA is run, each individual in the population is evaluated against the unknown environment. The fitness values are associated with the values of objective function.

D. Genetic Operators

Genetic operators drive the evolutionary process of a population in GA, after the Darwinian principle of survival of the fittest and naturally occurring genetic operations. The most widely used genetic operators are reproduction, crossover and mutation.

To perform genetic operators, one must select individuals in the population to be operated on .The selection strategy is chiefly based on the fitness level of the individuals actually presented in the population. There are many different selection strategies based on fitness. The most popular is the fitness proportionate selection.

After a new population is formed by selection process, some members of the new populations undergo transformations by means of genetic operators to form new solutions (a recombination step). Because of intuitive similarities, we only employ during the recombination phase of the GA three basic operators: reproduction, crossover and mutation, which are controlled by the parameter p_r, p_c and p_m(reproduction probability, crossover probability and mutation probability), respectively.

Let us illustrate these three genetic operators. As an individual is selected, reproduction operator only copy it form the current population into the new population (i.e., the new generation) without alternation.

The crossover operator starts with two selected individuals and then the crossover point (an integer between 1 and L-1, where L is the length of strings) is selected randomly. Assuming the two parental individuals are x_1 and x_2, and the crossover point is 5 (L=20). If

x_1=(01001|101100001000101)

x_2=(11010|011100000010000)

Then the two resulting offspring are

x'_1=(01001|011100000010000)

x'_2=(11010|101100001000101)

The third genetic operator, mutation, introduces random changes in structures in the population, and it may occasionally have beneficial results: escaping from a local optimum. In our GA, mutation is just to negate every bit of the strings, i.e., changes a 1 to 0 and vice versa, with probability p_m.

III. ENHANCED GENETIC ALGORITHMS

A. Hybrid Genetic Algorithm（HGA）

The GA and the traditional optimization algorithms have defects respectively. Most traditional optimization methods applied in engineering design require a better set of initial values for the design variables, and then converge rapidly to generate good results. However, most of those optimization algorithms face the same difficulties, such as a long trial-and-error process in finding a better set of initial design variables or slow convergence. The set of initial design variables is determined by engineering intuition in generally and different sets of initial design variables will in general give different optimum results. Therefore, how to select better initial values of the design variables is a critical step for those traditional methods. As for using the GA, it has the advantage of working in a random population. Even though a GA can locate the solution in the whole domain, it does not solve constraint problems easily, especially for exact constraints. In order to overcome these difficulties, a new hybrid optimization procedure, which combines the GA with traditional optimization methods, is presented in this study. In the first step of the procedure, the GA is applied to provide a set of initial design variables, thereby avoiding the trial process; thereafter, traditional algorithms are employed to determine the optimum results. This hybrid algorithm, which can be termed a *Hybrid Genetic Algorithm* (HGA), is more effective than traditional algorithms.

The flow chart of HGA is described as follows:

B. Interval Genetic Algorithm（IGA）

Optimization algorithms have been well developed in engineering for a long time; however, most optimization algorithms deal with methods with which to derive the design variables for an optimal design. Nevertheless, it usually is not easy to manufacture exact design variables in engineering, because of measurement inaccuracies or errors within the manufacturing process itself. Furthermore, to exact manufacturing is often more costly. Interval optimization is

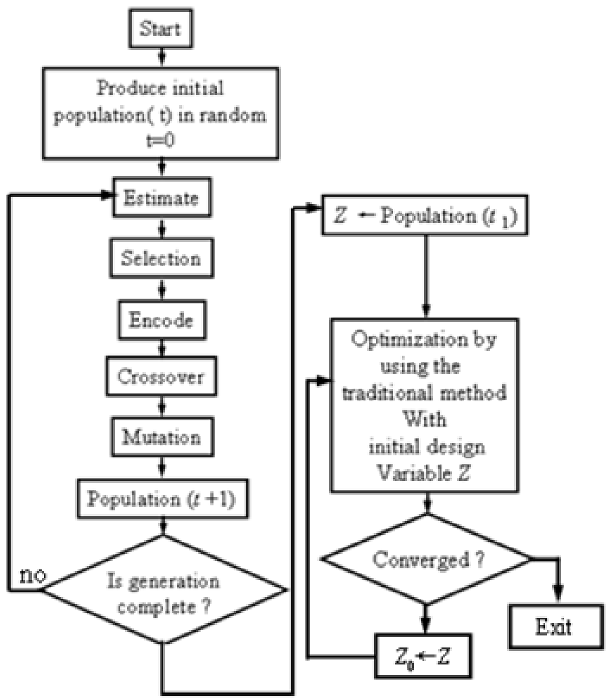

Fig.1 The flow chart of HGA

one of the algorithms that can overcome these difficulties. In the traditional interval optimization, the interval analysis is presented in the interval optimization process. The purpose of interval analysis is to provide upper and lower bounds on the effect of all such errors on a computed quantity. A complex interval can be a rectangle or a circle in the complex plane; or intervals on magnitude and phase can be used.

As the above statements indicate, the interval analysis is necessary and important for most interval optimizations. By using the interval analysis, it is easy to understand the relationship between the system performances and system parameters. But the interval differential formulation sometimes is not easy to be determined, especially with complicated systems. In this chapter, a new interval optimization method, called an *Interval Genetic Algorithm* (IGA), is proposed. With IGA, what matters is that optimum interval parameters can be derived. Additionally, not only interval analysis can be excluded in the process of optimization, but maximization of the optimum interval design scope can also be achieved.

The flow chart of IGA is described as follows:

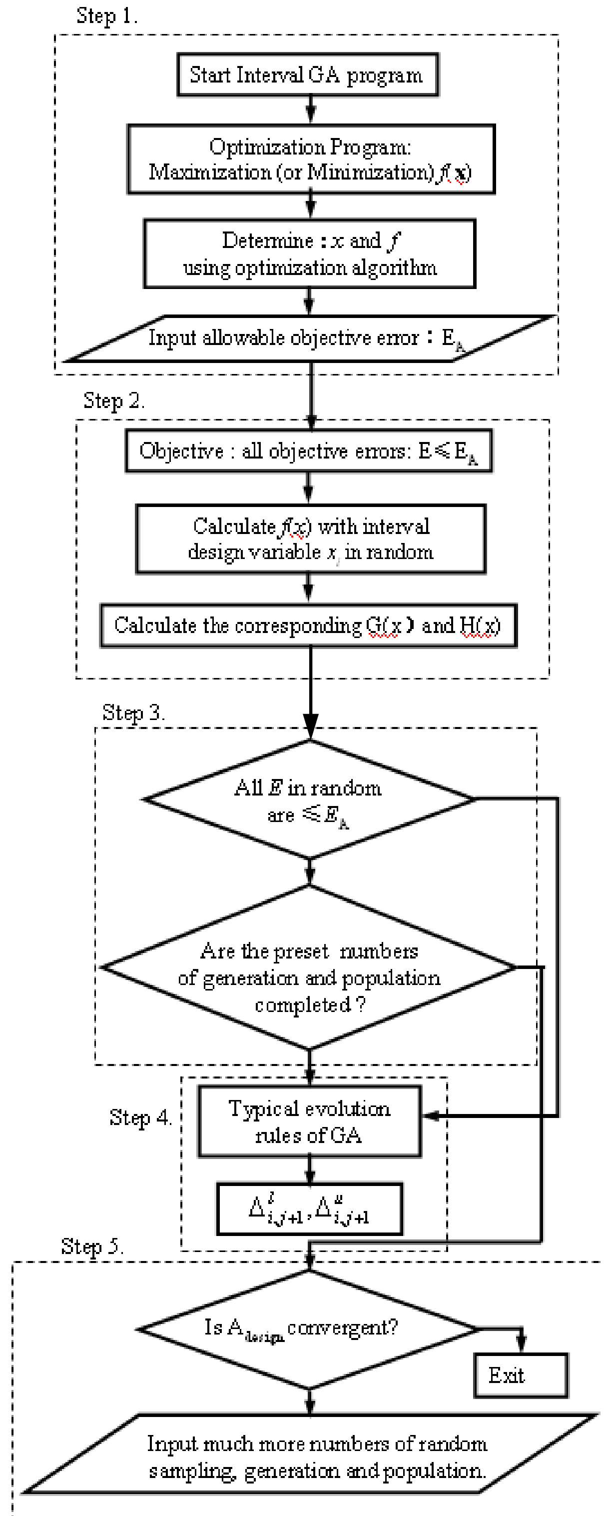

Fig.2 The flow chart of IGA

C. Hybrid Interval Genetic Algorithm (HIGA)

With the advancement of the computer ability, the interval analysis and interval optimization are respected and applied in all respects during the recent years, such as mathematics, biochemistry, engineering and so on. In engineering, the interval optimization is extensively applied in structure design. Notwithstanding the interval optimization has been applied in engineering for a long time, the interval analysis and mathematical calculating are indispensable. However, not only the differential formulation of complicated system is difficult to be determined, but also the complicated system sometime is not easy to be formulated. For the original IGA in the above section, the objective error E is calculated

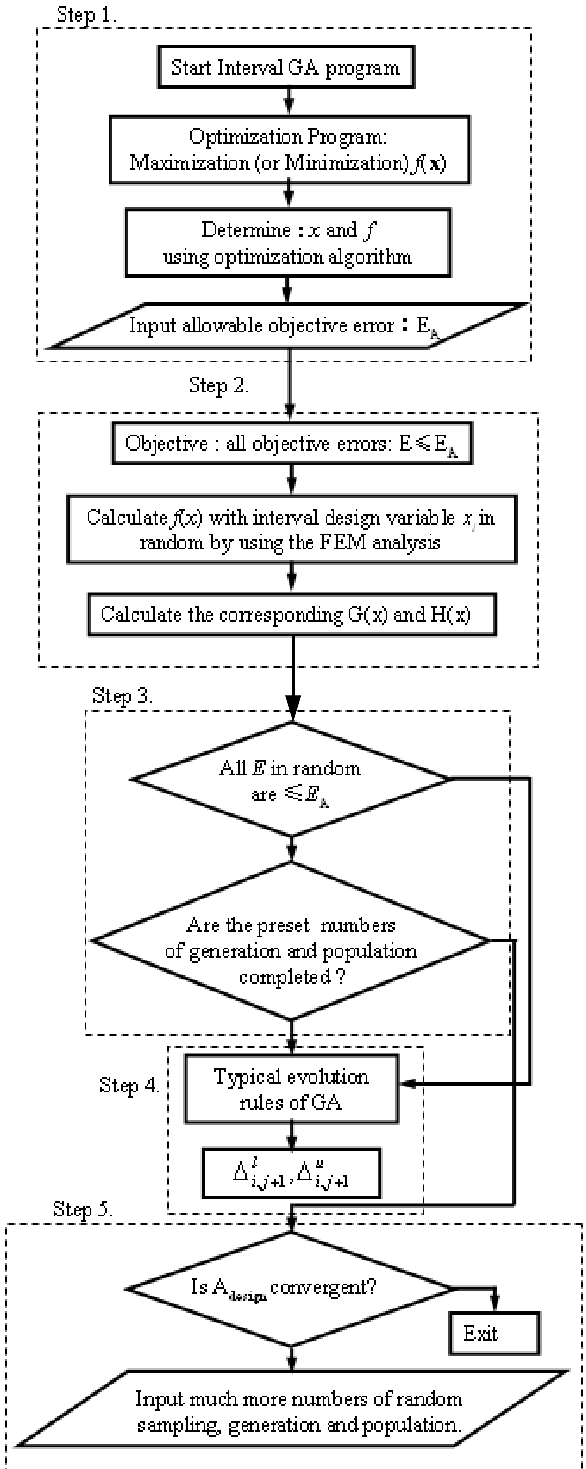

Fig.3 The flow chart of HIGA

from the formulated equation. In fact, it is easy to describe and formulate the system equations for an uncomplicated system, but complicated systems sometimes are not easy to achieve. Furthermore, the simplified system equation sometimes is difficult to determine the accurate solutions. To overcome those difficulties, the technology combines the IGA with Finite Element Method (FEM) software for the interval optimization is presented in this section.

IV. Numerical Examples

Example 1: Simple global optimization problem of the polynomial.

One global optimization problem is presented to prove the ability of the HGA and which is shown in Fig.4 (a).

$$f(x,y) = -20\frac{\sin\sqrt{0.1+(x-4)^2+(y-4)^2}}{\sqrt{0.1+(x-4)^2+(y-4)^2}}$$

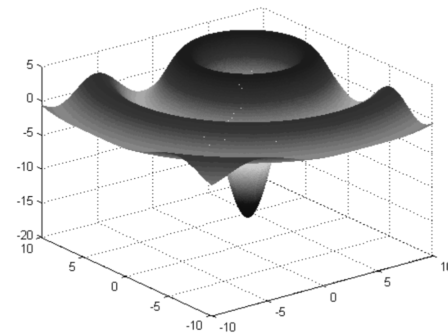

Fig.4 (a) the global optimization problem

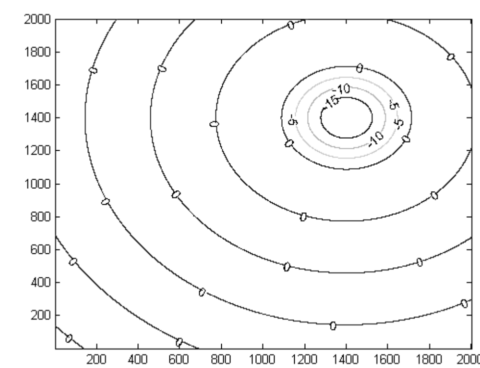

Fig.4 (b) Optimization with the HGA

The global minimum is at (x_1, x^2)=(4, 4) with the objective f^* =−19.6683. As the results show in Fig.4 (b), the optimization approach using the SQP got one local optimum value f (4, −3.75) =−2.5662. Because of the random initial value are inferior and the SQP can not determine the global optimum value with the inferior initial value. This example shows that the HGA can determine the global optimum value effectively.

Exampl2: Single-objective interval optimization of the polynomial.

One two-dimensional polynomial function is presented in this example, which is expressed as the following:

$$f(x,y) = 3(1-x)^2 e^{-x^2-(y+1)^2} - 10\left(\frac{x}{5} - x^3 - y^5\right)e^{-x^2-y^2}$$

This is shown in Fig.5 (a) and the contour map of this polynomial function in the design scope is shown in Fig.5 (b).

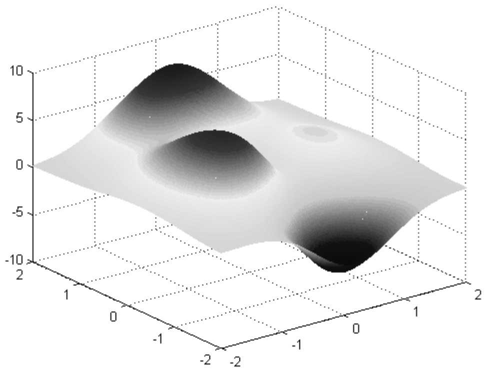

Fig.5 (a) *the interval optimization of the polynomial.*

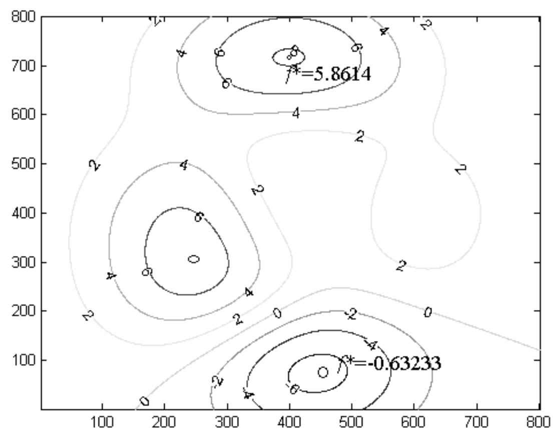

Fig.5 (b) Optimization with the IGA

Exampl3: A frame structure optimization using hybrid Interval Genetic Algorithm

The weight of the framed structure shown in Fig. 6 is to be minimized with stress and displacement constraints. Discrete values considered for this example are taken from the set of box-section steel (GB707-88) ***D***={25a, 25b, 25c, 28a, 28b, 28c, 32a, 32b, 32c, 36a, 36b, 36c, 40a, 40b, 40c} (type). Young's modulus is specified as E=200 GPa; the stress limit is [σ] =100 MPa; the displacement limit at each node is [δ]=20mm; the material density is ρ =7.8 g/cm^3. Two-load case is applied to the framed structure. Load case 1: q_1=10 KN/m, q_2=20KN/m, q_3=0; Load case 2: q_1=10 KN/m, q_2=20KN/m, q_3=15KN/m.The results are given in Table 3.

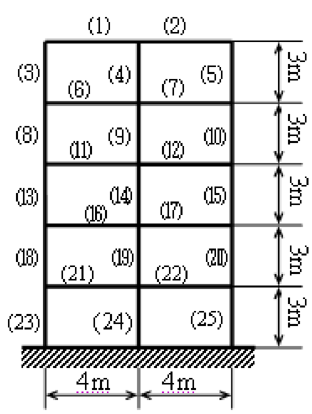

Fig.6 framed structure

Table 1 Comparison of designs framed structure

	HGA	IGA	HIGA
A_1	28c	32a	28c
A_2	36a	36a	32b
A_3	25a	25a	28c
A_6	36a	36a	32c
A_7	25a	25a	25b
g_1	-3.10	-7.43	-2.80
g_2	-3.54	-4.12	-2.55
g_3	-15.90	-13.98	-17.24
g_6	-7.85	-8.12	-3.38
g_7	-25.7	-26.12	-6.10
g_9	-10.3	-11.2	-9.28
W	1410	1480	1393

V. CONCLUSIONS

Three kinds of enhanced genetic algorithm are presented to overcome different engineering problems. In HGA, the genetic algorithm is applied to provide a set of initial design variables, thereby avoiding the trial process; and another optimization algorithm is employed to determine the final optimum results. This new interval optimization procedure is referred as the Genetic Algorithm and denominated as an *Interval Genetic Algorithm* (IGA). As opposed to former interval optimization algorithms, interval analysis can be excluded in this interval optimization process. In addition to the IGA, the *Hybrid Interval Genetic Algorithm* (HIGA) even combines the IGA with the Finite Element Method (FEM). This hybrid algorithm can exclude equations formulation and interval analysis, and determines the optimum interval parameters.

REFERENCES

[1] Guo Pengfei, Han Yingshi, "Chaotic genetic algorithm for structural optimization with discrete variables", *Journal of Liaoning Technical University*, 2007, 26(1), pp. 68-70

[2] Guo Pengfei, Han Yingshi, "An imitative full-stress design method for structural optimum design with discrete variables", *Engineering Mechanics*, 2003, 16(5), pp. 95-99

[3] Guo Pengfei, Han Yingshi, Wei Yingzi. "An Imitative Full-stress Design Method for Structural Optimization with Discrete Variables", *Engineering Mechanics*, 2000, 17(1),pp:94-98

[4] Arumugam, M. S., Rao, M.V.C., Palaniappan, , "New hybrid genetic operators for real coded genetic algorithm to compute optimal control of a class of hybrid systems," Applied Soft Computing Journal, 2005, 6(1), pp. 38-52

[5] Andrzej, P., "Optimization of mechanical structures using interval analysis," Computer Assisted Mechanics and Engineering Sciences, 2000, 7(4), pp. 699-705

[6] Chen, S. H. and Wu, J.,, "Interval optimization of dynamic response for structures with interval parameters," Computers and Structures, 2004,82(1), pp. 1-11

[7] Liu, J. L., "Intelligent genetic algorithm and its application to aerodynamic optimization of airplanes," AIAA Journal, 2005, 43(3), pp. 530-538

[8] McWilliam, S., "Anti-optimization of uncertain structures using interval analysis," Computers & Structures, 2001,79(2), pp. 421-430