International Journal of Hybrid Information Technology
Vol.9, No.3 (2016), pp. 125-132
http://dx.doi.org/10.14257/ijhit.2016.9.3.12

Improved Multi-objective Optimization Evolutionary Algorithm on Chaos

Xue Ding[1] and Chuanxin Zhao[2]

[1]School of Enrolment and Job-guidance Center, Yunnan Normal University, Kunming, 650500, China
[2]Information Center, Yunnan Transportation Department, Kunming, 650031, China
Corresponding Author: Xue Ding, School of Enrolment and Job-guidance Center, Yunnan Normal University, Kunming, 650500, China

Abstract

In this paper, chaos theory and the traditional multi-objective optimization evolutionary algorithm is put forward, "Chaos-based multi-objective evolutionary algorithm", combines a variety of optimization strategies. The traditional multi-objective evolutionary algorithm for repeating individual causes of variation is based on chaotic analysis of multi-objective evolutionary algorithm and demonstration. According to the characteristics of chaotic map tent, NSGA-II algorithm in this paper on the basis of chaotic map was proposed based on chaotic tent initialization and chaotic mutation multi-objective evolutionary algorithm. The original NSGA-II algorithm is improved, and the introduction of adaptive mutation operator and a new crowding distance is calculated and applied to the design of the algorithm. Analysis and experimental results show that these methods can better improve the distribution of population performance.

Keywords*: chaos, multi-objective optimization, evolutionary algorithm*

1. Introduction

In order to overcome randomly generated initial population is uneven, the proposed method using real coding based on chaotic initialization. Zhang Xi and others on the real-coded methods are analyzed [1], overall, the real coding has the following characteristics: First, the real number as the initial population, the encoding process is simple. Second, the real coding can be eliminated in the process of ordinary binary code "Hamming" cliff problem. Finally, the real-coded easy to control, especially for chaotic initialization and chaotic mutation. Population initialization is based on the scope of the problem and constraints. The proposed algorithm based on the triangular tent map：

$$p(k+1)=\begin{cases} 2p(k), 0\le p(k)\le 1/2 \\ 2-2p(k), 1/2\le p(k)\le 1 \end{cases} \tag{1}$$

1.1 Chaotic Mutation

This paper proposed multi-objective evolutionary algorithm the mutation operator is also chaotic mapping by chaotic mutation Tent achieved.

Mutation step is set, the production of expression also have Tent chaotic map given.

$$\delta(k+1)=\begin{cases} 2\delta(k), 0\le \delta(k)\le 1/2 \\ 2-2\delta(k), 1/2\le \delta(k)\le 1 \end{cases} \tag{2}$$

ISSN: 1738-9968 IJHIT

Chaotic mutation expression:

$$c_k = p_k + (p_k^u - p_k^l)\delta_k \tag{3}$$

c_k is offspring. p_k is parent, p_k^u is the upper bound of the parent component. p_k^l is a lower bound. δ_k is a small step generated by chaotic sequences.

1.2 Improved Calculation of Crowding Distance

In the multi-objective evolutionary algorithm, usually more than one objective function optimization. How to use a reasonable objective function to rate the individual is a serious problem. Deb in the NSGA-II, the proposed use of crowding distance (Crowing distance) to the individual to choose from. Assuming a multi-objective optimization problem with two goals and, in one species crowding distance of individuals is a long distance quadrilateral with a wide distance. Here the individual with the representative of the population crowding distance for the individual in a certain target value, the crowding distance:

$$P[i]_{distance} = (P[i+1].f_1 - P[i-1].f_1) + (P[i+1].f_2 - P[i-1].f_2) \tag{4}$$

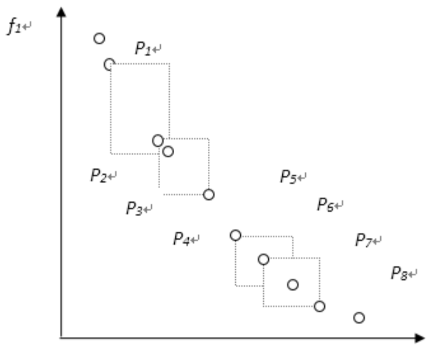

Figure 1. Deb's Crowding Distance

In Figure 1, P_1 , P_2 very close, one should be deleted, but due to congestion from the great, have been retained. And p_5, p_6, p_7, p_8 distribution is very uniform, but because of crowding distance is too small to be removed. Shihua Wen proposed a threshold based on an improved strategy [2].If less than a threshold value, the retention of individuals; the other hand, out of the individual. Specific strategies for improvement:

First find out the current layer of the elite two extreme points in the collection, find the distance between them, in which the threshold distance given by the following formula:

$$\delta = dist/(2\times popsize) \tag{5}$$

Crowding distance based threshold strategy shown to improve:

In this evaluation, the current uneven end (such as concave or convex), may be such a great influence on the results. This article proposes another new crowding distance is calculated.

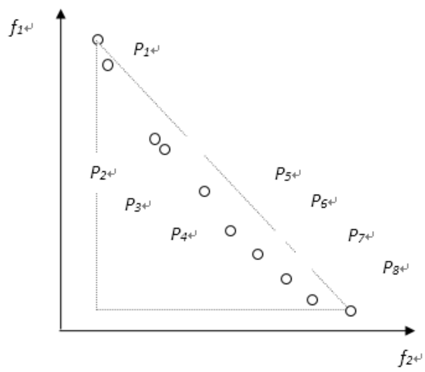

Figure 2. Threshold-based Improvement Strategies Crowding Distance

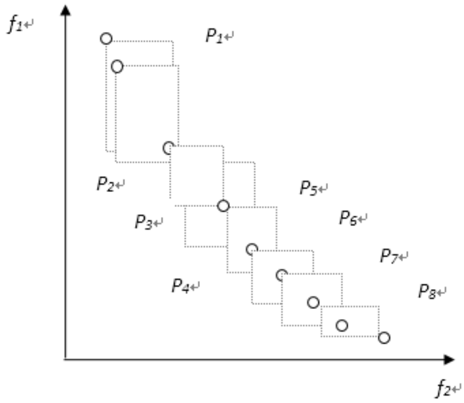

Figure 3. Threshold-based Improvement Strategies Crowding Distance

Figure 3 .Like the original idea, first find the front end of each individual crowding distance, and then calculate the average crowding the front-end distance of the individual.

$$\bar{d} = \sum_{i=1}^{popsize} P[i]_{distance} / popsize \tag{6}$$

The distance can be defined as:

$$d_i^{'} = P[i]_{distance} / \bar{d} \tag{7}$$

1.3 Dynamic Mutation Probability

Evolutionary algorithm mutation operator is the most important part. Good mutation operator can directly affect the performance of the algorithm. Therefore, the introduction of dynamic variation of the variance in order to better mediate (fine turning) effect. This variation means, first proposed by the Srinvivas in his method of p can change with the change of fitness. Specific algorithm is as follows [3]:

$$p = \begin{cases} k(f_{\max} - f)/(f_{\max} - f_{avg}),\ f \geq f_{avg} \\ k',\ f < f_{avg} \end{cases} \tag{8}$$

Here, $f_{\max}$ is the largest population fitness value, f is the fitness of the individual to choose the value, f_{avg} for the average fitness value. Mutation probability k, k 'dynamic adjustment.

Variation of this approach be a good method, but due to multi-objective evolutionary algorithm, the comparison is not between individuals according to the size of the fitness function to compare the way it carried out in accordance with disposable, so this method in a multi-objective evolutionary algorithm cannot be put to good use. The proposed crowding distance is calculated. This paper: based on crowding distance of the dynamic mutation probability, specifically described as follows:

$$p = \begin{cases} k(d - d_{\min})/(d_{avg} - d_{\min}), d \le d_{avg} \\ k', d > d_{avg} \end{cases} \tag{9}$$

Where, d for the need of individual variation in the crowded distance, d_{min} is the smallest population crowding distance, d_{avg} average for the population crowding distance. The meaning of these formulas is that when crowding is less than the average crowding distance, when, according to the probability of dynamic variation. When the crowding distance is greater than the average crowding distance, when the individual in order to prevent excessive dispersion, should be set to a small mutation probability k'.

1.4 Mutation Probability based on the Number of Iterations

Hashem also made a running time of algorithm with dynamic mutation of the algorithm [4]. With some ideas of the method, the paper also proposes a method based on variation of the number of iterations.

Fitness value calibration to solve two problems:

Early in the evolution of the probability of using relatively large chaotic mutation. The randomness of chaos and ergodicity of variation in decision-making space for a wide range of search, to avoid local optima. This can better maintain the population diversity. When the algorithm reaches a certain evolutionary stage of the process of calculation, the algorithm gradually converges to the global optimal solution, because the groups are close to the value of individual fitness, in order to prevent the individual is still a wider range of swing, it should reduce the mutation rate, relying onclose to Pareto optimal binary cross-surface slow.

In this paper, this variation formula:

$$p = (n - i)/n \times k \tag{10}$$

Among them, the objective evolutionary algorithm iterations n, i for the current iteration number, k is the mutation probability parameter.

1.5 Algorithm Performance Evaluation

In the multi-objective evolutionary algorithm, a fast solution to find the distribution are set to evaluate the importance of multi-objective evolutionary algorithm standards. Deb presented the performance evaluation of the five principles of MOEA [5].

The domain of the function values should be [0,1], so relatively easy to compare between different generations.

The true value of the objective function is known, so in theory can explain the value of the algorithm can be calculated.

To evaluate the function and evolution of algebra should be proportional, so that one can use the same algorithm to different Algebra.

The evaluation function should be applied to multiple targets, that is, the algorithm can adapt to different dimensions on the comparison.

Evaluation of the algorithm should try to lower the complex.

Here the introduction of the three test functions ZDT1, ZDT2, ZDT3 shown in Table 1: Test function list

Table 1

Test function	Definition	Constraints
ZDT1: Has a convex Pareto front	$f_1(x_1)=x_1$ $g(x_2,...,x_m)=1+9\sum_{i=2}^{m}x_i/(m-1)$; $h(f_1,g)=1-\sqrt{f_1/g}$;	$m=30$, $x_i\in[0,1]$ Pareto optimal front is formed $g(x)=1$
ZDT2: Non-convex Pareto front	$f_1(x_1)=x_1$ $g(x_2,...,x_m)=1+9\sum_{i=2}^{m}x_i/(m-1)$ $h(f_1,g)=1-(f_1/g)^2$	$m=30$, $x_i\in[0,1]$ Pareto optimal front is formed $g(x)=1$
ZDT3: Represent discontinuous features, it is not continuous prominent parts:	$f_1(x_1)=x_1$ $g(x_2,...,x_m)=1+9\sum_{i=2}^{m}x_i/(m-1)$ $h(f_1,g)=1-\sqrt{f_1/g}-(f_1/g)\sin(10\pi\cdot f_1)$	$m=30$, $x_i\in[0,1]$ Pareto optimal front is formed $g(x)=1$

1.6 Algorithm for the Evaluation Function

Evaluation process in the algorithm, using Zitzler and Deb [6] jointly proposed test function. Function in this group test, each test function is defined as including the three functions *f*, *g*, *h*:

Minimize $T(x)=(f_1(x_1), f_2(x_2))$

Satisfy $f_2(x)=g(x_2,...,x_m)h(f_1(x_1),g(x_2,...,x_m))$

As $x=(x_1,...,x_m)$

In this equation is the first decision variable is the remaining variables, equations and parameters values.

1.7 Algorithm Evaluation Criteria

Multi-objective evolutionary algorithm is currently evaluating a variety of criteria, listed below are several methods used to determine set of multi-objective evolutionary algorithm for solving the distribution and convergence.

(1) Error rate (Error Ratio, ER).Error rate is Coello Coello [7] proposed a multi-objective evolutionary algorithm solution set coverage approach. Its meaning can be understood as algorithms calculate the distribution of individual rates in the real Pareto front. The mathematical expression is:

$$ER=\sum_{i=1}^{n}e_i\bigg/n \tag{11}$$

e_i in the Pareto front where the number of vectors, n is the total number of vectors.

(2) Population from the (Generational Distance, GD).Van Veldhuizen's population from [8] is the Pareto front in the real desires of the individual and the algorithm the distance between the solution set. The mathematical expression is:

$$GD = \frac{(\sum_{i=1}^{n} d_i^p)^{1/p}}{n} \tag{12}$$

e_i in a true front-end solutions focused on the individual and the distance between an individual is the total number of individual solution set.

(3) Individual space (Spacing, SP).Schott presented the concept of individual space [9].It said the concentration of individuals in the solution space distribution.

$$SP = \sqrt{\sum_{i=1}^{n} (\bar{d} - d_i)^2} \div (n-1) \tag{13}$$

Where d-is the average distance, di is the distance between two adjacent individuals, n the total number of individuals for the solution set.

1.8 Experimental Results and Analysis of Algorithm Performance

In the experiment, the following four methods were compared: including NSGA-II (Non-dominated Sorting Genetic Algorithm-II) and ε-MOEA (ε-Multiobjective Evolutionary Algorithms) [10]. And the proposed non-dominated sorting based on chaotic evolutionary algorithm CNSGA (Chaos Non-dominated Sorting Genetic Algorithm, CNSGA) algorithm. And the addition of dynamic variation and crowding distance algorithm. Chaos-based evolutionary algorithm for adaptive non-dominated sorting CANSGA (Chaos Adapted Nondominated Sorting Genetic Algorithm, CANSGA)

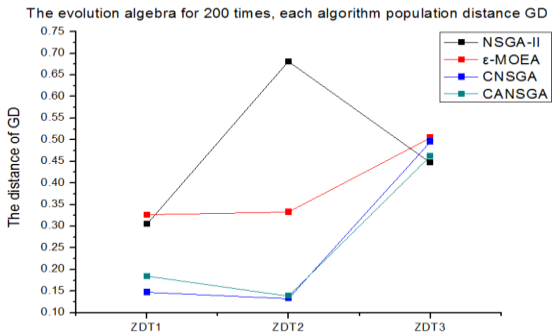

Figure 4. The Evolution Generation is 200 Times the Distance of Each Population GD Algorithm

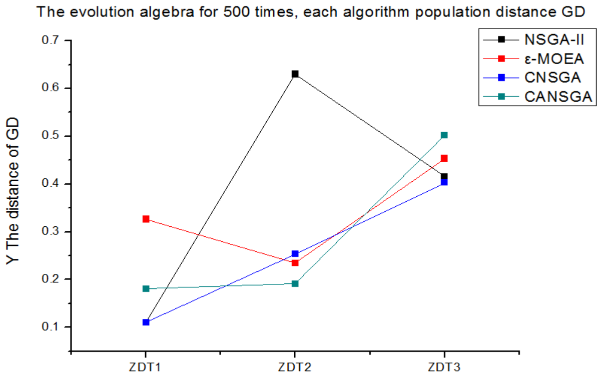

Figure 5. Evolution Generation is 500 Times the Distance of Each Population GD Algorithm

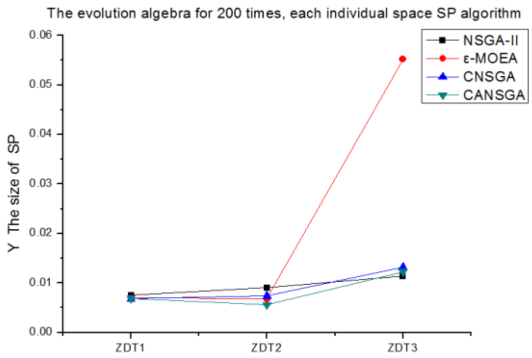

Figure 6. Evolution Generation is 200 Times, Each Individual Space SP Algorithm

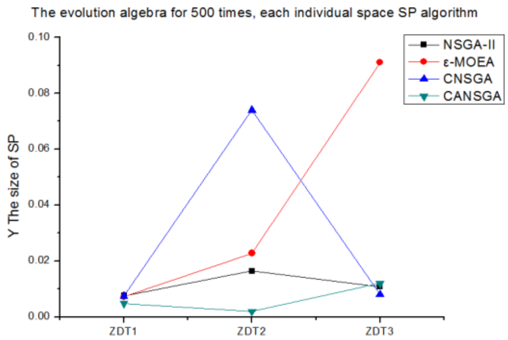

Figure 7. Evolution Generation is 500 Times, Each Individual Space SP Algorithm

GD said the Pareto front obtained by experiment to the real Pareto front of the distance from the above experimental results, when the evolution generation is 200 generations, the proposed algorithm and chaos optimization algorithms have a relatively good performance, but the effect is notprominent. When the evolution generation is 500 generations, the proposed algorithm in the three test functions have a relatively good performance. Especially in the ZDT2 and ZDT3 in the results obtained the minimum, which show that the algorithm has good convergence.

SP said the Pareto front obtained by experiments the crowding distance between individuals, with smaller values, indicating that the distribution of Pareto front more uniform between individuals. When the evolution generation is 200 generations, the proposed algorithm's performance is very good. When the evolution generation is 500 generations, the proposed algorithm and chaos optimization algorithm has a distinct advantage.

2. Conclusion

This paper is designed and implemented a dynamic chaos genetic algorithm. Specific work includes the following three aspects: 1, based on chaotic sequence initialization 2, the design of chaotic mutation 3, an improved calculation of crowding distance. The results show that the NSGA-II with the classical and the ε-MOEA compared to ordinary chaos-based multi-objective evolutionary algorithm and the Pareto front in the convergence of the uniform distribution of relatively well in terms of performance, and improved multi-objective evolutionary algorithm Chaos able to further improve the solution quality. And in the evolution of algebra to achieve less good results.

Acknowledgment

The author wishes to thank the helpful comments and suggestions from his teachers and colleagues in intelligent information processing key laboratory of colleges and universities in Yunnan province and also thank Enrollment and Job-guidance Center to provide hardware. This study is supported by the study fund of the common construction of build intelligent information project.

References

[1] Z. Xi, Z. Quanshou, "Study of empty flow distribution optimization based on real coding genetic algorithm, vol. 26, no. 11, **(2006)**, pp. 130-138,144.
[2] W. Shihua, "Study of mutation operator in the multiobjective evolutionary algorithm", Xiangtan Univers ity, **(2009)**.
[3] M. Srinvivas and L. M. Patnaik, "Adaptive probabilities of crossover and mutation in genetic algorithms", IEEE Tran SMC, vol. 24, no. 4, **(1994)**, pp. 656-666.
[4] M. M. A. Hashem, M. Watanabe and K. Izumi, "Evolution strategy a new time-variant mutation for fine local tuning", Proceedings of the 36th SICE Annual Conference. International Session Papers, **(1997)**, pp. 1099-1104.
[5] K. Deb and S. Jain, "Running Performance Metrics for Evolutionary Multi-Objective Optimization", KanGAL Report No.2002004, **(2002)**.
[6] E. Zitzler, K. Deb and L. Thiele, "Comparison of Multi-objective Evolutionary Algorithms: Empirical Results", Evolutionary Computation, vol. 8, no. 2, **(2000)**, pp. 173-195.
[7] C. A. Coello, "An Updated Survey of GA-Based Multi-objective Optimization Techniques", ACM Computing Surveys, vol. 32, no. 2, **(2000)**, pp. 109-143.
[8] V. Veldhuizen, A. David and G. B. Lamont, "Evolutionary Computation and Convergence to a Pareto Front", The Genetic Programming Conference, **(1998)**, pp. 221-228.
[9] J. R. Schott, "Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization", Department of Aeronautics and Astronautic, Massachusetts Institute of Technology Massachusetts, **(1995)**.
[10] K. Deb, M. Mohan and S. Mishra, "A Fast Multi-objective Evolutionary Algorithm for Finding Well-Spread Pareto-Optimal Solutions", KanGAL Report No.2003002, **(2003)** February.