ISSN: 2664-9640

Multi-objective evolutionary algorithm based on dynamic scale factor

Suping Liu

College of Software Engineering, Guangdong university of science and technology, Dongguan, Guangdong 532000, China

Abstract

This paper proposes a high-dimensional multi-objective evolutionary algorithm based on dynamic scale factors. The dynamic scale factor can dynamically adjust the influence ratio of the new distance factor and the crowding density distance in the evolution process, thereby balancing the diversity and convergence of the evolution process, by solving classic multi-objective problems and comparing other classic multi-objectives The algorithm verifies the rationality of the new algorithm.

Keywords

Multi-objectives; dynamic scale factor; distance factor.

1. Introduction

The research of multi-objective evolutionary algorithm started relatively late, and it is still in the stage of rapid development. The earliest multi-objective optimization problem can be traced back to 1772. At that time, Franklin put forward the problem of multiple objective conflicts. However, most scholars believe that the earliest proponent of the multi-objective optimization problem is the French economist V. Pareto. It should have been proposed by him in 1896. He summarized many difficult-to-comparable problems as multi-objective optimization problems. In 1951, TCKoopmans realized the multi-objective optimization problem of production activities, and proposed an important concept of multi-objective optimization—Pareto optimal solution (non-inferior solution); in 1968, Z.Johnsen put forward the multi-objective problem more comprehensively. Vigorously promote the development of multi-objective optimization problems. However, the multi-objective optimization problem has experienced almost 70 years of development from V. Pareto's proposal to Z. Johnsen's comprehensive summary[1].

In 1984, David Schaffer first proposed the combination of evolutionary algorithms and MOPs (Schaffer, 1985). In 1989, David Goldberg introduced Pareto theory from economics to multi-objective evolutionary algorithms. A large number of multi-objective evolutionary algorithms have been proposed one after another. For example, the first generation of multi-objective evolutionary algorithms such as MOGA [2], NSGA[3] etc., and the second generation of multi-objective evolutionary algorithms. Target evolution algorithm PAES [4], PESA-II [5], SPEA[6] and NSGA-III [7] and other algorithms.

Based on this, this paper proposes a high-dimensional multi-objective evolutionary algorithm based on dynamic scale factors. The dynamic scale factor can dynamically adjust the influence ratio of the new distance factor and the crowding density distance during the evolution process, thereby balancing the diversity and convergence of the evolution process, by solving classic multi-objective problems and comparing other classic multi-objectives Algorithm to verify the performance of the algorithm.

ISSN: 2664-9640

2. 2. Evolutionary algorithm

Evolutionary algorithm is a population-based search algorithm that requires multiple iterations to solve, and includes operations such as selection, crossover, and mutation. The algorithm flow is shown in Table 1 [2]:

Table 1 Basic flow of evolutionary algorithm

Basic flow of evolutionary algorithm
Input: population size N, crossover probability Pc, mutation probability Pm and fitness calculation rules
Output: individual optimal solution
Step 1 Initialize. Randomly generate N individuals as the initial population X(0); set the evolution algebra counter t=0;
Step 2 Individual evaluation. Calculate the fitness of each body in X(t);
Step 3 Population evolution (multiple evolution strategies can be applied);
Step 4 Choose an operation. Select M/2 pairs of parents from X(t) (M≥N);
Step 5 Cross operation. Perform a crossover operation on the selected M/2 pairs of parents according to the crossover probability Pc to form M intermediate individuals;
Step 6 Mutation operation. Perform mutation operations on these M individuals according to the mutation probability Pm to obtain M candidate sub-individuals;
Step 7 Select (Children). Select N individuals from M candidate sub-individuals according to a certain selection strategy to form a new generation of population X(t+1);
Step 8 Terminate the inspection. According to the fitness calculation rules, judge whether the termination condition has been reached, if yes, output the optimal solution in X(t+1) and terminate the calculation; otherwise, set t=t+1 and go to step 3).

3. Multi-objective algorithm based on dynamic scale

In order to be able to comprehensively consider the convergence and diversity of the algorithm, therefore, the design of the individual ranking function is very important. The sorting function designed in this subject is as follows:

$$R_i^j = (1-\rho_j)D_i + \rho_j C_i \tag{1}$$

The sorting function is divided into two parts. The former is the generation distance value of the first individual in the population (the newly designed distance factor in this paper), which is used to measure the contribution of the i-th individual in the population to the convergence of the algorithm, and the latter is the crowding distance of the first individual in the population Value, used to measure the contribution value of the i-th individual in the population to the diversity of the algorithm. It is a variable scale operator that represents the weight value of the first iteration in the evolution process. Through the right design, the population can be based on the entire evolution process. Different characteristics embodied in different stages to carry out dynamic control.

According to the Schema theorem in evolutionary computing and the basic idea of evolutionary algorithm design, applying it to multi-objective evolutionary algorithm shows that since the initial population of evolutionary algorithm is generated randomly, in the initial stage of evolution, the large population in the population Some individuals become dominated individuals with a high probability. In order to allow the algorithm to converge in a more correct direction, a larger weight value should be assigned to the generation distance. With the continuous progress of the evolutionary process, the elite selection strategy used in the evolutionary algorithm will gradually reduce the number of dominated individuals. Therefore,

the weight value assigned to the generation distance at this time should also be reduced accordingly. When the evolutionary algorithm reaches the final stage, most of the population has become non-dominated individuals. In order to further increase the selection pressure of the algorithm, a larger weight value should be set for the crowded distance at this time. Therefore, the variable scale operator in this paper is set as follows:

$$\rho_j = \frac{j}{G_{\max}} * (\rho_{\max} - \rho_{\min}) + \rho_{\min} \tag{2}$$

Where ρ=[0.1, 1], and $G_{\max}$ represents the maximum number of iterations set by the algorithm. It can be seen from the above formula that the scale operator is a time-varying variable that increases with the number of iterations of the algorithm. The dynamic change of the scale operator can better control the contribution value of the generation distance factor and the crowded distance factor in different stages of evolution in the ranking function.

4. Experimental results and analysis

In order to verify the superiority and rationality of the improved multi-objective evolutionary algorithm, this part verifies the analysis by solving some classic multi-objective test functions. The experimental process is to combine the improved sorting algorithm with the multi-objective evolutionary algorithm to solve the multi-objective optimization problem, and compare it with some classic multi-objective evolutionary algorithms. This part of the experiment uses a total of 5 algorithms for comparison. The purpose of the experiment is to judge whether the results obtained by the algorithm in this paper have better results than other comparison algorithms in terms of evaluation indicators. The comparison algorithms used in this paper are NSGAII, SPEA2, and MOEA/D. In the evolutionary algorithm, the corresponding parameters need to be set. This paper uses the classical parameter values, the mutation probability is set to 0.7, the crossover probability is set to 0.4, the population size is set to 250, and the number of solutions does not exceed 1000* the dimension of the individual. Set the dynamic scale factor as ρ_{max}=0.9, ρ_{min}=0.1.

This paper uses the classic test function set DTLZ function. DTLZ is a set of test functions constructed by Deb et al. The naming is composed of the first letters of the family names of the four authors. The function set contains eight test functions. In this paper, DTLZ1 and DTLZ2 are selected to test the algorithm of this paper [8]. The three evaluation indicators of generation distance, anti-generation distance and hypervolume are used to evaluate the performance results. In the experiment, a variety of Multi-objective algorithms are compared. The experimental results are shown in Table 2 and Table 3. It is easy to see that the method in this paper has very obvious advantages.

Table 2 DTLZ1 experimental results under the four goals

Indexes		NSGAII	SPEA2	MOEA/D	Proposed algorithm
IGD	Best	0.245077	0.796523	0.126005	0.110639
	Worst	7.863667	24.928042	0.143029	**0.112134**
	Mean	1.476302	7.747862	0.134259	**0.111490**
	Std	1.855256	6.542981	0.003782	**0.000382**
GD	Best	0.405617	4.334574	0.014722	0.011188
	Worst	15.752902	34.207384	0.018043	**0.011374**
	Mean	5.684001	20.364070	0.016803	**0.011296**
	Std	4.238780	7.482417	0.000716	**0.000046**
	Best	0.707693	0.131014	0.833524	**0.863969**

HV	Worst	0.000000	0.000000	0.805563	**0.862935**
	Mean	0.236934	0.006551	0.819961	**0.863578**
	Std	0.242061	0.028554	0.007082	**0.000241**

Table 3 DTLZ2 experimental results under the four goals

Indexes		NSGAII	SPEA2	MOEA/D	Proposed algorithm
IGD	Best	0.153077	0.131165	0.155225	0.101023
	Worst	0.182311	0.148021	0.176936	**0.101153**
	Mean	0.164517	0.140376	0.165685	**0.101089**
	Std	0.006318	0.005289	0.005707	**0.000030**
GD	Best	0.009813	0.010776	0.009092	0.008238
	Worst	0.013263	0.016028	0.010260	0.008251
	Mean	0.011506	0.012578	0.009688	0.008244
	Std	0.000843	0.001571	0.000296	**0.000004**
HV	Best	0.450529	0.512570	0.496347	**0.582924**
	Worst	0.400628	0.458952	0.444223	**0.582675**
	Mean	0.426630	0.487321	0.474193	**0.582835**
	Std	0.012736	0.013508	0.014921	**0.000051**

5. Conclusion

The dynamic scale-based multi-objective optimization algorithm proposed in this paper can effectively optimize the target, and the experimental results show that this method can effectively improve the performance of the algorithm and is better than several commonly used multi-objective algorithms.

References

[1] Penna P, Prada A, Cappelletti F, et al. Multi-objectives optimization of Energy Efficiency Measures in existing buildings[J]. Energy and Buildings, 2015, 95: 57-69.
[2] Murata T, Ishibuchi H. MOGA: Multi-objective genetic algorithms[C]//IEEE international conference on evolutionary computation. 1995, 1: 289-294.
[3] Deb K, Agrawal S, Pratap A, et al. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II[C]//International conference on parallel problem solving from nature. Springer, Berlin, Heidelberg, 2000: 849-858.
[4] Knowles J D, Corne D W. M-PAES: A memetic algorithm for multiobjective optimization [C]// Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No. 00TH8512). IEEE, 2000, 1: 325-332.
[5] Corne D W, Jerram N R, Knowles J D, et al. PESA-II: Region-based selection in evolutionary multiobjective optimization[C]//Proceedings of the 3rd annual conference on genetic and evolutionary computation. 2001: 283-290.
[6] Mendoza F, Bernal-Agustin J L, Domínguez-Navarro J A. NSGA and SPEA applied to multiobjective design of power distribution systems[J]. IEEE Transactions on power systems, 2006, 21(4): 1938-1945.
[7] Ishibuchi H, Imada R, Setoguchi Y, et al. Performance comparison of NSGA-II and NSGA-III on various many-objective test problems[C]//2016 IEEE Congress on Evolutionary Computation (CEC). IEEE, 2016: 3045-3052.

ISSN: 2664-9640

[8] Deb K, Agrawal S, Pratap A, et al. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II[C]//International conference on parallel problem solving from nature. Springer, Berlin, Heidelberg, 2000: 849-858.