A Multi-objective Dynamic Differential Evolution Algorithm Based on Population Strategy

Jing An[1, 2]

*[1]Yancheng Institute of Technology,
Yancheng 224051, Jiangsu, China
[2]Hohai University, Nanjing 210098, Jiangsu, China*

Sen Xu*

*Yancheng Institute of Technology,
Yancheng 224051, Jiangsu, China
Corresponding author, email: 27894737@qq.com

Abstract
Multi-objective optimization is one of the main research fields in terms of optimization problem. As one of today's most effective random optimization algorithms, differential evolution algorithm is an effective tool to solve the multi-objective optimization problem. On behalf of the population evolution of the whole solution set, the differential evolution algorithm searches in the internal parallel way to be able to get multiple non-dominated solutions by optimizing the operation. Based on the “greed” selection mechanism and the “non-dominant” sorting idea of the population strategy, this paper focuses on the characteristics of multi-objective optimization problem and proposes a multi-objective dynamic differential evolution (MODDE) algorithm for solving such kind of problem. The experimental results show that the MODDE algorithm can solve to get approximate optimal solution with good convergence and dispersion can deal with continuous and discrete Pareto frontier multi-objective problems in various forms.
Key words: MULTI-OBJECTIVE OPTIMIZATION, DIFFERENTIAL EVOLUTION ALGORITHM, POPULATION STRATEGY

1. Introduction

Multi-objective optimization problems widely exist in practical engineering application and scientific researches, which are a kind of very important, difficult and complex problems to solve. Since the 1960s, the multi-objective optimization problem has attracted the attention of more and more researchers from different backgrounds, and this is because the multi-objective optimization problem is very common and plays a vital role in real life[1]. Under same conditions, the processing with optimized technology will realize significant effects in such respects as the system efficiency increase, lower energy consumption, reasonable use of resources and the improvement of economic benefit etc. Generally speaking, optimization problems in the science and engineering practices are mostly multi-objective optimization and decision making problems. It is a topic with very scientific research value and practical significance to solve the multi-objective optimization problems[2].

In terms of the multi-objective optimization, as there exist non-comparison and mutual contradiction between objectives, it is often very difficult to make each sub-objective achieve the optimal state at the same time. The reason lies in that the fact that one certain solution may be optimal in terms of one objective but worst in terms of another, and what the multi-objective optimization problem searches for is usually a group of equilibrium solutions, i.e. Pareto optimal solution[3]. Franklin put forward the problem of how to coordinate multi-objective contradictions in 1772, which can be considered as the earliest origin of multi-objective problem. In 1896, the French economist V. Pareto firstly proposed multi-objective programming problem in the study of economic balance and introduced the concept of the so-called Pareto optimal solution, and then spread such concept, and the generally accepted international multi-objective optimization problem was firstly put forward by him. In 1963, L.A. .Zadeh put forward multi-objective control problem from the angle of cybernetics. In 1968, Z.Johnsen study carried out the study report of multi-objective decision-making problem, and such systematical study report is a landmark in the multi-objective optimization field and promotes the rapid development of the multi-objective optimization problem. Algorithms for solving multi-objective optimization problems include the non-Pareto algorithm and Pareto algorithm, and non-Pareto algorithm does not directly use the basic concept of Pareto optimization, and has the characteristics of high efficiency and is easy to implement. But it cannot produce some parts of Pareto optimal front end, while, the Pareto algorithm can make the whole population approach the Pareto optimal front end by using the non-dominated sorting and selection. Differential evolution (DE) algorithm is a novel heuristic intelligent search algorithm[4]. The differential evolution algorithm was originally used to solve the Chebyshev polynomial problem, but Storn etc. finds that DE is also effective in solving complex optimization problems. Differential evolution algorithm simulates the learning process of the individual’s collecting information in the population, and the reason why it is suitable to solve the multi-objective optimization problems lies in that a set of Pareto optimal solutions can be obtained in one operation process through the group information sharing operation, and besides, it is unnecessary to carry out the sorting of math in terms of multi-objective optimization problem. For most objective functions and constraint conditions, the differential evolution algorithm can be adopted to get solutions[5].

This paper first elaborates the multi-objective optimization problem, and presents the mathematical description of the multi-objective optimization problem, and then introduces the principle and evolution of differential evolution algorithm. Based on the above, the multi-objective dynamic differential evolution algorithm (MODDE) for solving multi-objective optimization problems is designed based on the “greed” selection mechanism and the non-dominant sorting idea of the population strategy. Finally the experimental simulation and analysis are discussed.

2. Mathematical Description of Single-objective Optimization Problem and Multi-objective Optimization Problem

With strong application, the optimization theory and technology is a discipline based on math, and the optimized quantity which only needs one objective function to express is known as the single-objective optimization problem, and the one with more than one objective needing being optimized is known as multi-objective optimization problem(referred to as MOP). In terms of multi-objective optimization problem, each sub-objective is mutually contradictory, and the improvement of one sub-objective may cause the performance degradation of another or the other few sub-objectives, that is to say, it is impossible to make many sub-objectives reach the optimal value at the same time[6]. The core of the multi-objective intelligent algorithm is to coordinate the relationship between the objective functions, and make each objective function seek the bigger (or smaller) optimal solution set as far as possible. One solution may be the best in terms of one certain objective, but the worst in terms of the other one. There is no solution which is optimal in terms of all objectives. This solution is usually referred to as the non - dominated solution or non inferior solution or Pareto solution, which was referred to by Vilfredo Pareto in 1896. Compared with the single objective optimization problem, the multi-objective planning corresponds to multiple optimal solutions. How to make the decision makers choose the required solution and how to construct the optimal solution set of multi-objective optimization problems are problems existing in the study of multi-objective programming[7].

2.1. Mathematical Description of The Single-objective Optimization

Suppose $S \subseteq R^n$ as the search space of n dimension, and $X = (x_1, x_2, \cdots x_n) \in S$ one decision vector, and $X \in [X_{\min}, X_{\max}]^n$ the boundary constraint. Suppose $f(X)$ as the objective function and take the minimization for example, then single-objective optimization can be described as:

$$\begin{aligned} \min imize \quad & f(X) \\ & X \in [X_{\min}, X_{\max}]^n \end{aligned} \quad (1)$$

2.2. Mathematical Description of Multi-objective Optimization

Suppose $S \subseteq R^n$ as the search space of n dimension, $X = (x_1, x_2, \cdots x_n) \in S$ one decision vector, and $X \in [X_{\min}, X_{\max}]^n$ the boundary constraint. Suppose $F(X) = (f_1(X), f_2(X), \cdots f_{n_f}(X))$ as the objective vector containing n_f objective functions, and take the minimization for example, then multi-objective optimization can be described as:

$$\begin{aligned} \min imize \quad & F(X) \\ & X \in [X_{\min}, X_{\max}]^n \end{aligned} \quad (2)$$

Generally, for multi-objective optimization problem, there exists no only global optimal solution, so the process of solving multi-objective optimization problems is the process of seeking Pareto solution set. Research of multi-objective programming algorithm mainly focuses on the following three aspects: first, the acquisition of optimal solution set, second, the quality problem of the solution, and third, the management of constraint functions [8].

3.Differential Evolution Algorithm

The basic differential evolution algorithm is the algorithm based on the candidate scheme population, and it is realized by combining existing schemes in the population by using simple mathematical formula, and the scheme is searched in the whole search space. If the new scheme is improved, it will be accepted, otherwise discarded, and this process is repeated until the satisfactory solution is found. Due to such advantages of DE algorithm as easiness to understand and implement, it attracts much attention and has been widely used after the proposal. DE evolutionary operators include mutation, crossover and selection.

(1) Population initialization

Initial population $X(0) = \{x_1^0, x_2^0, \cdots, x_{NP}^0\}$ is randomly produced in the solution space, in which, $x_i^0 = \{x_{i,1}^0, x_{i,2}^0, \cdots, x_{i,D}^0\}$ is used to represent the individual solution of number i population. Each weight of the individual can be produced according to the following formula:

$$x_{i,j}^0 = x_{j,\min} + rand(x_{j,\max} - x_{j,\min}) \quad (3)$$

In which, $x_{j,\max}$ and $x_{j,\max}$ are respectively the upper and lower bound of number j dimension in the solution space[9].

(2) Mutation operation

For any objective vector x_i in the parental population, the mutation vector v_i can be generated according to the following formula:

$$v_i = x_i + F(x_{i_2} - x_{i_3}) \quad (4)$$

$$v_i = x_{i_1} + F(x_{i_1} - x_{best}) + F(x_{i_2} - x_{i_3}) \quad (5)$$

In which, $F > 0$ is the real constant controlling the differential mutation, and i_1, i_2, i_3 are individuals randomly selected from the whole population, and x_{best} is the best solution found so far[10].

(3) Crossover operation

The aim of differential evolution algorithm crossover operation is to improve the variety of population individual through the random recombination of each dimension weight of mutation vector v_i and objective vector x_i. New crossover vector is generated according to the following formula:

$$v_{ij} = \begin{cases} v_{i,j}, & if\,(U < CR) \\ v_{i,j}, & otherwise \end{cases} \quad (6)$$

In which, CR is the crossover probability and U is the variable randomly generated among 0 and 1.

(4) Selection operation

The selection operation of differential evolution algorithm is one “greed” selection mode. If and only if the fitness value of new vector individual u_i is better than that of the objective vector individual x_i, u_i will be accepted by the population. Otherwise, x_i will still be reserved in the population of the next generation and continue to serve as the objective vector in the iterative computation of the next generation to implement the mutation and crossover operation. Assume the optimization problem as $\min f(x)$, then the selection operation can be described according to the following formula:

$$x_i = \begin{cases} v_i, & if\,(f(v_i) < f(x_i)) \\ v_i, & otherwise \end{cases} \quad (7)$$

In which, fit the fitness value. In terms of DE, the greed selection method is adopted to guarantee that the population of the next generation contains the vector with better fitness value.

The individual behavior of differential evolution is mainly manifested in the differential mutation operator and crossover operator. As a random search algorithm simulating natural evolution phenomenon, differential evolution algorithm is likely to realize the global optimal search, but it also has the premature disadvantage. Population has more scattered random configuration at the beginning, but with the process of evolution, the population distribution density among each generation is higher, and the information exchange is reduced gradually, thus making the global optimization ability decrease gradually[11]. Illustration of mutation and crossover operations is shown in Fig.1.

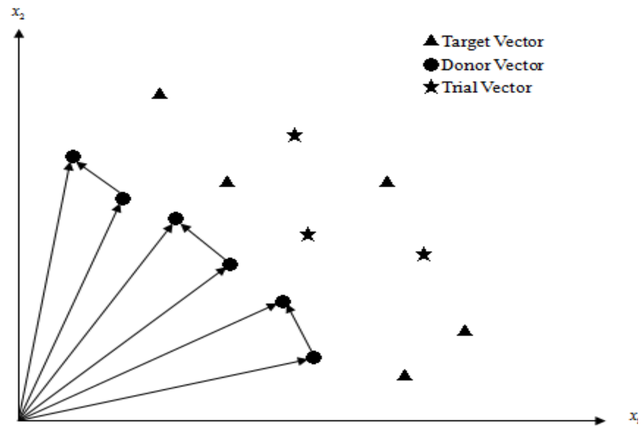


Figure 1. Illustration of mutation and crossover operations

4. MODDE Algorithm Design

In order to keep the population structure in the process of evolution and ensure the normal operation of the algorithm, when the givenevolution algebra is limited, the evolution multi-objective optimization algorithm can't fully guarantee that enough elite solutions are achieved to be stored in the external archive, and even sometimesit occurs that elite solution number in external archive is less than the population size[12]. In order to avoid possible problems occurring in the above-mentioned external archive strategy, based on "greed" selection mechanism and the non-dominant sorting idea, this paper proposes a non-elite archive multi-objective differential evolution algorithm of the dynamic and update population to be used to solve the multi-objective optimization problem. Differential evolution algorithm using "greed" selection mechanism and the process of population dynamic update is the individual domination selection process[13]. The algorithm design is as follows:

(1) When the objective individual $X_{li}^t(i=1,\dots,N_l)$ dominates the test individual U_{li}^t, X_{li}^t directly enters into the next generation without updating the current population P_l^t.

(2) When the test individual U_{li}^t dominates the objective individual X_{li}^t, the test individual U_{li}^t replaces the objective individual X_{li}^t and enters into the current population P_l^t.

(3) When the test individual U_{li}^t and objective individual X_{li}^t are different from each other and do not dominate each other, the test individual U_{li}^t will also enters into the current population P_l^t but rank the last in the current population, and the current population P_l^t size will correspondingly add 1. It is worth noting that this new individual entering the current population will take part in the later evolution operation of all objective individuals, but it will no longer serve as objective individual to carry out the mutation, crossover and selection operation in the current population[14].

(4) The objective individuals involved in the evolution in this current population are still the first N_l number of individuals in the current population P_l^t. By the dynamic update, the fine filial generation (test individual U_{li}^t) generated from the evolution of the front individual X_{li}^t in the current population P_l^t will immediately affect the evolution of the back individual $X_{li}^t(i<j\leq N_l)$, thus contributing to increase the convergence rate of the algorithm[15].

(5) When the first N_l number of individuals in the population P_l^t is all evolved, each individual will get the corresponding non-dominant level ND_l after the non-dominant sorting is conducted on all individuals of the population P_l^t that finally acquired after the dynamic update.

(6) If the final acquired population size is larger than the given population, each individual's crowded entropy CD_l will be further calculated.

(7) Then, select the sub-generation population P_l^{t+1} entering into the evolution iteration of the next generation according to individual's non-dominant level ND_l and crowded entropy CD_l. Firstly the level $ND_l=1$ individual will be considered and selected from large CD_l to small CD_l, and when all $ND_l=1$ individuals are selected, then the level $ND_l\geq 2$ individual will be

considered and selected according to the same manner till N_l number of individuals are selected.

5. MODDE Algorithm Testing

We select the multi-objective optimization testing problem, i.e. DTLZ testing function to verify the validity of above proposed MODDE algorithm, the following is the introduction to the problem and Pareto front.

$$\begin{array}{ll}
f_1(x)=\cos(0.5\pi x_1^a)\cdots\cos(0.5\pi x_{m-2}^a)\cos(0.5\pi x_{m-1}^a)(1+g(x)) & \\
f_2(x)=\cos(0.5\pi x_1^a)\cdots\cos(0.5\pi x_{m-2}^a)\sin(0.5\pi x_{m-1}^a)(1+g(x)) & [0,1]^n \\
f_3(x)=\cos(0.5\pi x_1^a)\cdots\sin(0.5\pi x_{m-2}^a)(1+g(x)) & n=10 \\
\vdots & m=3 \\
f_m(x)=\sin(0.5\pi x_1^a)(1+g(x)) &
\end{array}$$

(8)

Figure 1 is the Pareto optimal front of such problem, and its numerical representation is shown in Formula (8). Pareto entropy’s change is caused by the domination of the old solution in the Pareto front end by the new solution acquired by the evolution algorithm, or the replacement of the old solution with poor quality. In terms of the multi-objective evolution algorithm adopting the external file to store Pareto optimal solution set, it is generally thought that the quality of Pareto solution with larger individual density is poor. During the external file update process, Pareto new solution with smaller individual density is often adopted to replace Pareto old solution with larger individual density, thus enhancing the variety similar with Pareto front end, including the distribution uniformity and ductility. Therefore, the dominant and diverse details in the evolution iteration process can be speculated by evaluating Pareto differential entropy in the process of evolution, so as to estimate the evolution state of the multi-objective evolution algorithm, thus providing feedbackto the dynamical adjustment of the evolution strategy. But to divide the boundary value of Pareto differential entropy with different evolutionary status is the key factor of this method, and the unreasonable boundary value will lead to the misjudgment of the evolutionary status, consequently, will be likely to induce the adoption of contrary evolution strategy by the evolution algorithm. Pareto front of DTLZ testing function adopted by this paper is shown as Figure 2.

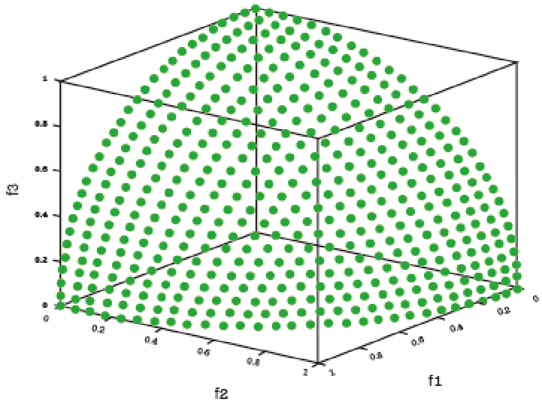
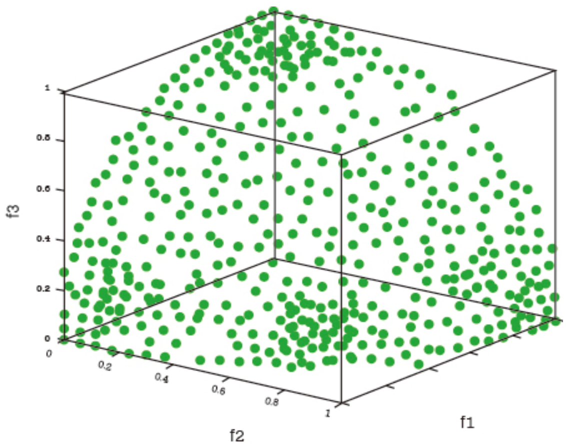
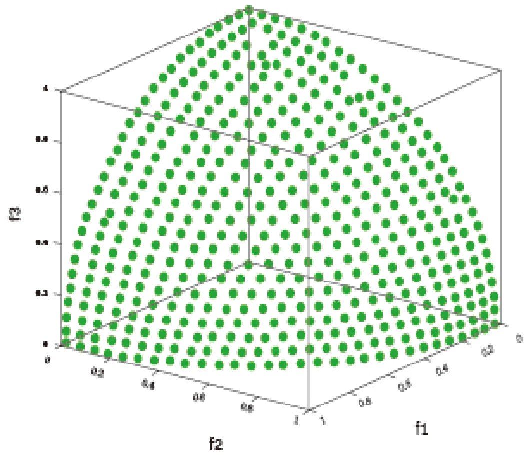


Figure 2. Pareto front of DTLZ testing function

The optimization problem of such test function is the constrained multi-objective optimization problem, and the variable number is more and the solution is very complex. In order to evaluate the efficacy of the MODDE algorithm, the approximate Pareto boundary acquired by solving DTLZ problem. Here the comparison with NSGA-II algorithm is shown in Figure 3.



(a) NSGA-II Algorithm



(b) MODDE Algorithm

Figure 3. Pareto front obtained by solving DTLZ problem

From the above Figure 3, there exists difficulty in NSGA-II in convergence process, and the approximate boundary cannot be converged. MODDE algorithm performs best, because the corresponding approximate boundary can accurately converge into one curve in 3D space. The optimal solution obtained by MODDE algorithm can well approach Pareto optimal solution, and also has a good variety and breadth, and all obtained optimal solutions very approach the constrained boundary, which is consistent with the relationship represented by Formula (8).

6. Conclusion

Evolution algorithm has been successfully applied in the multi-objective optimization field, and developed into a relatively hot evolutionary multi-objective optimization research direction. The improved differential evolution algorithm MODDE adopted in this paper solves multi-objective optimization problems, and designs a dynamic multi-objective differential evolution algorithm by keeping interacting populations, thus realizing the concurrent execution of the optimization tasks by the iteration. The experimental simulation proves that MODDE can effectively deal with large-scale multi-objective optimization problems.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under (Grant No. 61105057), Jiangsu Province Qing Lan Project, and Nature Science Foundation of the Jiangsu Higher Education Institutes of China under (Grant No.13KJB520024).

References

1. José M. Chaves-González, Miguel A. Pérez-Toledano (2015) Differential Evolution with Pareto Tournament for the Multi-objective Next Release Problem. *Applied Mathematics and Computation*, 252(1), p.p.1-13.
2. Yingguo Chen, Vladimir Mahalec, Yingwu Chen, Xiaolu Liu, Renjie He, Kai Sun (2015) Reconfiguration of Satellite Orbit for Cooperative Observation using Variable-size Multi-objective Differential Evolution. *European Journal of Operational Research*, 242(1), p.p.10-20.
3. José M. Chaves-González, Miguel A. Vega-Rodríguez (2014) DNA Strand Generation for DNA Computing by using a Multi-objective Differential Evolution Algorithm. *Biosystems*, 116(2), p.p.49-64.
4. Umesh Kumar Rout, Rabindra Kumar Sahu, Sidhartha Panda (2013) Design and Analysis of Differential Evolution Algorithm Based Automatic Generation Control for Interconnected Power System. *Ain Shams Engineering Journal*, 4(3), p.p.409-421.
5. Fábio Fabris, Renato A. Krohling (2012) A Co-evolutionary Differential Evolution Algorithm for Solving Min–Max Optimization Problems Implemented on GPU using C-CUDA. *Expert Systems with Applications*, 39(12), p.p.10324-10333.
6. Shivom Sharma, Gade Pandu Rangaiah (2013) An Improved Multi-objective Differential Evolution with a Termination Criterion for Optimizing Chemical Processes. *Computers & Chemical Engineering*, 56(13), p.p.155-173.
7. Musrrat. Ali, Patrick Siarry, Millie Pant (2012) An Efficient Differential Evolution Based Algorithm for Solving Multi-objective Optimization Problems. *European Journal of Operational Research*, 217(2), p.p.404-416.
8. J. Preetha Roselyn, D. Devaraj, Subhransu Sekhar Dash (2014) Multi Objective Differential Evolution Approach for Voltage Stability Constrained Reactive Power Planning Problem. *International Journal of Electrical Power & Energy Systems*, 59(7), p.p.155-165.
9. Ayong Hiendro, Tanjungpura (2011) Multiple Switching Patterns for SHEPWM Inverters Using Differential Evolution Algorithms. *International Journal of Power Electronics and Drive Systems*, 1(2), p.p.94-103.
10. Linda Slimani, Tarek Bouktir Algeria (2012) Optimal Power Flow Solution of the Algerian Electrical Network using Differential Evolution Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, 10(2), p.p.199-210.
11. Saber M. Elsayed, Ruhul A. Sarker, Daryl L. Essam (2014) A Self-Adaptive Combined Strategies Algorithm for Constrained Optimization Using Differential Evolution. *Applied Mathematics and Computation*, 241(15), p.p.267-282.
12. Adam P. Piotrowski (2014) Differential Evolution Algorithms Applied to Neural Network Training Suffer from Stagnation. *Applied Soft Computing*, 21(8), p.p.382-406.
13. Banaja Mohanty, Sidhartha Panda, P.K. Hota (2014) Differential Evolution

Algorithm Based Automatic Generation Control for Interconnected Power Systems with Non-Linearity. *Alexandria Engineering Journal*, 53(3), p.p.537-552.

14. Manohar Singh, B.K. Panigrahi, A.R. Abhyankar, Swagatam Das (2014) Optimal Coordination of Directional Over-Current Relays Using Informative Differential Evolution Algorithm. *Journal of Computational Science*, 5(2), p.p.269-276.
15. Ali R. Yildiz (2013) Hybrid Taguchi-differential Evolution Algorithm for Optimization of Multi-Pass Turning Operations. *Applied Soft Computing*, 13(3), p.p.1433-1439.