The Open Automation and Control Systems Journal, 2015, *7*, 338-345

Open Access

A Simple Pareto Adaptive ε-Domination Differential Evolution Algorithm for Multi-Objective Optimization

Yan Jingfeng[1,*], Li Meilian[1], Xu Zhijie[1] and Xu Jin[2]

[1]*School of Information Engineering, Xuchang University, Xuchang, Henan, 461000, China, P.R. China*

[2]*School of Computer, China University of Geosciences, Wuhan, Hubei, 430074, China, P.R. China*

Abstract: The two purposes of solving the multi-objective optimization problems are to get solutions close to the true Pareto front as much as possible and to obtain promising diversity. To meet these two demands, a new method is proposed in this paper, which has these characteristics: 1) it adopts the orthogonal design method with quantization technology to generate initial population whose individuals are scattered uniformly over the target search space. 2) it is based on an adaptive ε concept to obtain a good distribution along the true Pareto-optimal solutions. Experiments on five benchmark problems with different features indicate that the proposed method works well not only in diversity, but also in convergence when compared to other evolutionary algorithms.

Keywords: Multi-objective optimization, Differential evolution, Adaptive ε domination, Pareto optimal.

1. INTRODUCTION

In the natural sciences and social sciences, we hope to maximize efficiency and minimize costs, which is essentially a multi-objective optimization problem [1]. As each objective restricts each other through decision variables in multi-objective optimization problems, it will result in the loss of other target's performance to over-optimize one objective. Therefore, it is difficult to evaluate the pros and cons of the solutions of multi-objective optimization problem. There is no unique global optimal solution in multi-objective problem, instead, it is a collection of the optimal solution which is commonly known as the Pareto optimal solution set. It is different from single objective optimization problem which is clearly defined and has just single optimal solution. Elements in the Pareto optimal solution set are non-domination and not comparable, that is to say, when considering all targets, there does not exist a solution which is better than these solutions. The main task of the multi-objective algorithm is to find solutions with good convergence and diversity which meet the requirements of the representative.

In last decades, a lot of algorithms were introduced to address multi-objective optimization problem, for instance, aggregation approaches, VEGA algorithm [2], the lexicographic ordering, ε-constrains method, the target-vector method, NSGA [3], MOGA [4], and NPGA [5]. In the late 1990s, some improved algorithms were come out, such as: PAES [6], SPEA [7], NSGA-II [8] and NPGA [9]. Researches on multi-objective genetic algorithm design and theoretical in domestic also showed the situation to go out in the ascendant. Yuping Wang [10] and Sanyou Zeng [11] give a new method for solving multi-objective optimization respectively by combining orthogonal design with uniform design into genetic algorithm, which accelerate the convergence speed. Shihua Guan [12] introduces an Augmented Lagrangian multi-objective collaborative algorithm based on the ε-constraint method. Chuan Shi [13] proposes a quick multi-objective evolution algorithm based on domination tree, it assigns fitness through domination tree which results in a less comparison among individuals. Gong etc. [14] apply two-level orthogonal crossover operator to DE and select optimal individuals using statistical optimization which lead to a better robust. Maoguo Gong [15] generalizes the current trend of research on multi-objective optimization and then put forward their own views on the further development of multi-objective optimization.

Although there have been many evolution algorithms to solve multi-objective optimization problems, it is still a challenge to design an efficient and robust algorithm. To this end, the authors introduce a differential evolution algorithm associating with ε domination to solve multi-objective optimization, which is termed Paε-ODEMO. The algorithm has following characteristics:

(1) applying orthogonal design method to generate the initial population, which not only decreases the time consumption, but also makes the initial population of points scattered uniformly over the feasible solution space.

(2) using an Archive population to retain the obtained non-dominated solutions. What's more, an adaptive ε way is adopted to maintain the diversity and distribution of the archive population dynamically.

*Address correspondence to this author at School of Information Engineering, Xuchang University, Xuchang, Henan, 461000, P.R. China; Tel: 15637474243; E-mail: jeffery8224@126.com

1874-4443/15 2015 Bentham Open

Finally, experiments on five benchmark problems of diverse complexities have shown that the new approach is able to achieve comparable result in terms of convergence and diversity metrics when compared with several other state-of-the-art evolutionary algorithms.

The paper is organized in the following manner. Section 2 introduces some basic knowledge about multi-objective optimization and differential evolution algorithm. Section 3 presents the core idea of our algorithm Paε-ODEMO. Section 4 summarizes some experimental results on continuous benchmark functions and analyzes the results. Section 5 provides some brief conclusion.

2. MULTI-OBJECTIVE DIFFERENTIAL EVOLUTIONARY ALGORITHM

In this section, we mainly focus on introducing some prerequisite knowledge about multi-objective optimization and differential evolution algorithm. They are the foundation of Paε-ODEMO.

2.1. Relative Description on Multi-objective Optimization

Definition 1: (multi-objective optimization problem): without loss of the generality, a MOP with a set of n decision variables, a set of m constrains and a set of k objective functions can be described as follows:

$$\begin{aligned} &minimize: && \mathbf{y}=\mathbf{f}(\mathbf{x})=(f_1(\mathbf{x}),\ldots\ldots f_k(\mathbf{x})\ ) \\ &subject\ to: && \mathbf{e}(\mathbf{x})=(e_1(\mathbf{x}),\ldots\ldots,e_m(\mathbf{x})\ )\geq 0 \\ &where\quad : && \mathbf{x}=(x_1,x_2,\ldots\ldots,x_n)\in X \\ & && \mathbf{y}=(y_1,y_2,\ldots\ldots,y_k)\in Y \end{aligned} \tag{1}$$

where $\mathbf{x}$ is the decision vector, $\mathbf{y}$ is the objective vector. X denotes as the decision space, while Y means the objective space. Generally, for each decision vector, it satisfies the constrain functions.

Definition 2: (Pareto dominance) assuming objective vectors $\mathbf{x}=(x_1,\ldots\ldots x_k)$ and $\mathbf{y}=(y_1,\ldots\ldots y_k)$ are two feasible solutions. if and only if

$$\forall i\in 1,\ldots\ldots k \quad x_i\leq y_i \quad and \quad \exists i\in 1,\ldots\ldots k \quad x_i<y_i \tag{2}$$

which is said $\mathbf{x}$ Pareto dominate $\mathbf{y}$. denotes: $\mathbf{x}\prec\mathbf{y}$

Definition 3: (ε dominance) a given objective vector $\mathbf{x}=(x_1,\ldots\ldots x_k)$ is said to ε dominate another objective vector $\mathbf{y}=(y_1,\ldots\ldots y_k)$. if and only if

$$\forall i\in 1,\ldots\ldots k \quad (1\text{-}\varepsilon)x_i\leq y_i \tag{3}$$

which denotes: $\mathbf{x}\prec_{\varepsilon}\mathbf{y}$

Definition 4:(Pareto optimal)given a solution x which is Pareto optimal, if and only if

$$\neg\exists\ \mathbf{x}\in X \qquad makes \qquad \mathbf{x}\prec\mathbf{x}^* \tag{4}$$

Definition 5:(Pareto front) the Pareto font is the appearance formed by all Pareto-optimal solutions when mapped to the objective vector space. Its mathematical description is as follows:

$$POF=\{f(\mathbf{x})=(f_1(\mathbf{x}),\ldots\ldots,f_k(\mathbf{x})\,|\,\mathbf{x}\in X\} \tag{5}$$

2.2. Differential Evolution Algorithm

DE is a random-based heuristic search algorithm, it is a kind of evolution algorithm based on real-coding which is proven to be an effective technique to solve complex optimization problems. It has been developed to be a mature algorithm and its framework can be described as follows: (1) Parameter assignment and generating the initial evolution population, set the evolution generation t=0;

(2) Calculate the fitness of the initial population;

(3) Stopping the algorithm and output the result if the stopping condition is met, else jump to step (4);

(4) Apply crossover operator, mutation operator and selection operator to the population, and then generate the next generation;

(5) Calculate the fitness of the population;

(6) Let the generations t=t+1, then jump to step (3);

What is necessary to point out is that our algorithm employs a hybrid selection mechanism in which a random selection and an elitist selection are interleaved. To be specific, in the early evolution, individuals in Archive population have big difference with the Pareto optimal solutions, so they do little help to guide the evolution. At this moment, the algorithm selects individuals from evolution population directly. However, with the progression of evolution, individuals in Archive population will be closer and closer to the Pareto optimal set. At this time, individuals are chosen from Archive population randomly, namely:

$$individual=\begin{cases} random \qquad eval<(\lambda\times Max\_eval) \\ elitist,otherwise \end{cases} \tag{6}$$

3. PAε-ODEMO AND ITS CHARACTERISTICS

In this section, we will present the proposed algorithm Paε-ODEMO in detail. Compared with traditional evolution algorithm for multi-object, it has many improvements and characteristic.

3.1. Initial Population with Orthogonal Design

In traditional DE, the initial population is generated randomly, which often leads to an uneven distribution, and then, decreases the algorithm's ability to use the initial population. While the initial population generated by orthogonal design method [16] can have a better distribution and diversity along the decision vector space. Orthogonal initial population can not only improve utilization capacity of the initial population, but also accelerate the convergence speed. There are two main steps to generate the initial orthogonal population:

(1) Generating orthogonal array [17], which is a kind of Latin array, using $L_R(Q^C)$ to represent an array with Q level;

(2) Quantization of the search space [18]. assuming a variable xi has a boundary of [l;u], and then quantize it into Q level as follows:

$$a_i = l + (i-1)(u-1)/(Q-1) \qquad i = 1,2,\ldots\ldots Q \tag{7}$$

3.2. Adaptive ε to Maintain Diversity of the Archive

In the procedure of updating population in this algorithm, whenever a new offspring is produced, it should be compared not only with its parents, but also with the individuals in archive population. If the new offspring is better than some individuals in archive, then just delete those individuals who are dominated; if the new offspring and all individuals in archive are not dominated mutually, then add it to the archive; if it is dominated by someone individual in archive, then discard the new one. In this case, as the running of evolution, the size of archive population will be larger and larger, which will make it hard to store, meanwhile, the distribution of archive will be getting worse and worse (individuals within a region have a high density, but other regions, a small density). In order to obtain not only a better distribution, but also reducing the storage space, the algorithm uses adaptive ε grid to maintain the diversity of archive. When the size of archive population reaches a predefined value, we need to generate the grid and map all individuals in archive population to the grid, and then use grid method to update the archive population. In order to store these mapped individuals, we assign an identity array to each individual, where r represents the number of objectives. The formula to calculate the B is shown as follows:

$$B_i = (\text{int})((f_i - f_i^{\min})/\varepsilon_i) \qquad i = 1,\ldots\ldots,r \tag{8}$$

Where $f_i^{\min}$ represents the minimum of the ith objective, the individual with the minimum fitness is from archive, and it varies as the running of evolution. The ε_i means the tolerance of the i th objective, with ε_i we can map each individual into the grid easily.

In order to generate the grid we need to calculate the corresponding ε_i. Without loss of generality, assume that one objective function has a range of $1 \le f_i \le K$, then for an optimization problem with r objective, it will produce a number of *r* grid which can accommodate $(\frac{K-1}{\varepsilon})^r$ non-dominated solutions. For an evolution population with NP size, the algorithm hopes to produce NP non-dominated solutions, so it will be met;

$$NP = (\frac{K-1}{\varepsilon})^{r-1} \tag{9}$$

And for this reason, we can calculate each ε_i easily. The reason why the ε_i is adaptive is that in most multi-objective optimization problems, the range of an objective is unknown at the beginning, but it is essential to know its range when calculate ε_i. Here, the maximum and minimum for a certain objective function are obtained from the archive population, because the algorithm thinks that individuals in archive are optimal, although they are just temporary optimals. As the evolution progresses, the maximum and minimum of a certain objective function may change, so that we can adjust the ε_i through the changing of its range, thereby adjusting the location of individual mapped to the grid.

Algorithm 1 shows the procedure of our algorithm update archive population. When a new offspring is generated, we use (8) to calculate its identity array **Ba**. Obviously, if *Ba[i] <0*, the minimum of i^{th} objective will change and the offspring is out of the original grid. And then, we should adjust the grid to accommodate it. However, to avoid adjusting gird too frequently, we just regenerate grid when *Ba[i] < 3*, and then readjust the position of each individual in the grid.

4. SIMULATION RESULT

In this section, we present some benchmark functions and then conduct some simulations to verify the convergence and diversity of our algorithm. What's more, we also analyze the results.

4.1. Test Case

In order to test the Pareto front of the multi-objective problems with various features, for instance convex, concave, discrete, uniform distribution, we have adopted some standard test functions below:

Case 1:

$$ZDT1 \quad \begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)[1-(1-(x_1)/g(x))^{0.5}] \\ g(x) = 1 + 9(\sum_{n=2}^{n} x_i)/(n-1) \end{cases}$$

In ZDT1, the size of decision vector n is 30, for each variable $x_i \in [0,1]$, it has a convex Pareto optimal front.

Case 2:

$$ZDT2 \quad \begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)[1-(1-(x_1)/g(x))^{2}] \\ g(x) = 1 + 9(\sum_{n=2}^{n} x_i)/(n-1) \end{cases}$$

In ZDT2, the size of decision maker n is 30, for each variable $x_i \in [0,1]$, it has a concave Pareto optimal front.

Table 1. Comparison of convergence of each algorithm in five benchmark functions (bestsolutionifbolded).

Algorithm	ZDT1	ZDT2	ZDT3	ZDT6	DTLZ1
NSGA-II	0.000894	0.000824	0.043411	7.80680	3.75997
INSGA-II	0.00057	0.00027	0.00330		
SPEA2	0.023285	0.16762	0.18409	0.23255	3.235731
AEPSO	0.00100	0.00078	0.00462		
DEMO	0.00554	0.14345	0.07951	0.57403	0.06361
NPCA*	0.000236	0.0004428	0.000231	0.001752	0.004036
PBFO	0.0170	0.0058	0.0058	0.0014	NA
ε-DEMO	0.00476	0.01682	0.0086	0.61489	0.02364
Paε-ODEMO	**0.000187**	**0.000195**	**0.000226**	**0.001237**	**0.001911**

Case 3:

$$ZDT3 \quad \begin{cases} f_1(x)=x_1 \\ f_2(x)=g(x)[1-(1-\sqrt{x_1/g(x)}-x_1\sin(10\pi x_1)/g(x))] \\ g(x)=1+9(\sum_{n=2}^{n}x_i)/(n-1) \end{cases}$$

In ZDT3, the size of decision maker n is 30, for each variable $x_i \in [0,1]$, its Pareto optimal front is discrete and concave.

Case 4:

$$ZDT6 \quad \begin{cases} f_1(x)=1-\exp(-4x_1)\sin^6(6\pi x_1) \\ f_2(x)=g(x)[1-(f_1(x)/g(x))^2] \\ g(x)=1+9[(\sum_{n=2}^{n}x_i)/(n-1)]^{0.25} \end{cases}$$

In ZDT6, the size of decision maker n is 10, for each variable $x_i \in [0,1]$, it has a concave non-uniform distribution of Pareto optimal front.

Case 5:

$$DTLZ1 \quad \begin{cases} f_1(x)=0.5x_1x_2(1+g(x_M)) \\ f_2(x)=0.5x_1(1-x_2)(1+g(x_M)) \\ f_3(x)=0.5x_1(1-x_1)(1+g(x_M)) \\ g(x_M)=100[|x_M|+\sum_{x_i\in X_M}(x_i-0.5)^{0.5}-\cos(20\pi(x_i-0.5))] \end{cases}$$

In ZDT6, the size of decision maker n is 12, for each variable $x_i \in [0,1]$, it is a three objective optimization problem.

4.2. Evaluation Indicators

(1) Convergence γ, denotes as . It was proposed by Van Veldhuizen and Lamont in 1998, used to represent approximation degree of the obtained optimal set Q and the real Pareto front PF, which is calculated as follows:

$$\gamma=\sum_{i=1}^{|Q|}d_i/(|Q|) \tag{10}$$

Where d_i means the smallest Euclidean distance between individual *i* in Q and individuals in PF, the smaller the value of γ, the better the algorithm performs. It's one of principle indicators.

(2) Δ metric. Deb etc., proposed the indicator in 2002, which is used to evaluate the distribution along the Pareto front, its formula is described as follows:

$$\Delta=(d_f+d_l+\sum_{i=1}^{|Q-1|}|d_i-\bar{d})/(d_f+d_l+(|Q-1|)\bar{d}) \tag{11}$$

In the formula above, d_i means the Euclidean distance between two neighboring individuals in Q, $\bar{d}$ is the average of all d_i, d_f and d_l represents the distance between the boundary solutions of Q and Pareto front which are mainly used in evaluating bi-objective optimization problem. The smaller the Δ value, the better the algorithm performs, especially when $\bar{d}=d_i$, Δ=0, at this time the solution obtained by this algorithm is distributed uniformly along the Pareto front.

However, this indicator has shown defects through analysis. For instance, it is mainly used to evaluate bi-objective optimization which has limitation to a certain extent. What's more, if there are only two boundary solutions in Q, at this time, Δ = 0, but it doesn't mean the algorithm is good. This indicator prefers to evaluate uniformity rather than distribution. Nevertheless, it is a valid indicator to evaluate performance of most algorithms.

Table 2. Comparison of diversity of each algorithm in five benchmark functions (bestsolutionifbolded).

Algorithm	ZDT1	ZDT2	ZDT3	ZDT6	DTLZ1
NSGA-II	0.463293	0.435112	0.575606	0.644477	0.95023
INSGA-II	**0.24073**	0.40057	0.56963		
SPEA2	0.472254	0.473808	0.606826	0.670549	0.8507665
AEPSO	0.55743	0.51901	0.55182		
DEMO	0.33648	0.68299	0.70669	0.8625	0.46383
NPCA*	0.402266	0.511157	0.580019	0.463253	NA
ε-DEMO	0.32187	0.36593	0.48575	0.51863	**0.43465**
Paε- ODEMO	0.288694	**0.304138**	**0.484418**	**0.248236**	0.520197

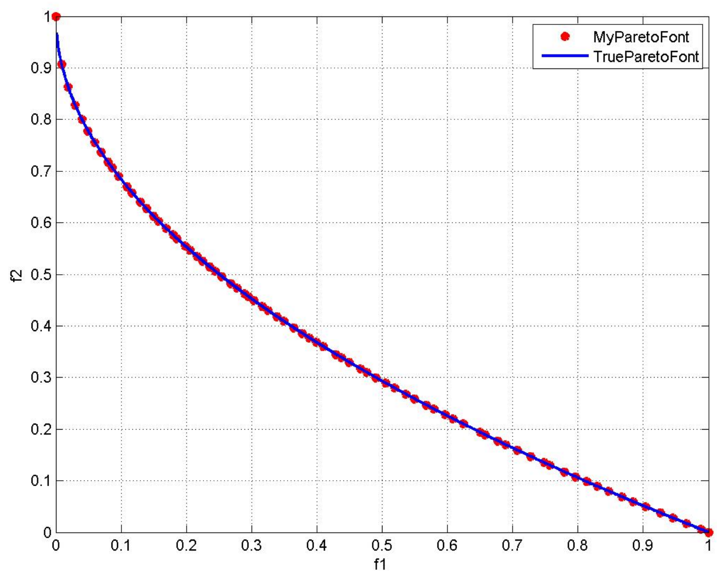

Fig. (1). Search trace to minimize 30-D ZDT1,it can get two boundary solutions easily.

4.3. Experiment Results and Analysis

The experiment is based on VC++6.0 platform, each algorithm runs 20 times independently, and then computes its average. In this algorithm, evolution parameters are set as follows: the size of population *NP=100*, the max fitness evaluations *Maxeval=25000*, scale factor *F=0.5*, selection control parameter λ=0.1, the index of orthogonal design *J=2*. For ZDT1, ZDT2, ZDT3, *Q=29*; for ZDT6 and DTLZ1, *Q=21*, the size of archive size NF=100 and crossover probability *CR=0*.9. In this paper, Paε-ODEMO is compared with several other state-of-the-art evolution algorithms: NSGA-II, SPEA2, INSGA-II [19], AEPSO [20], NPCA* [21], PBFO [22], DEMO [23], ε-DEMO [24], to verify its performance.

Seen from Table **1**, the Paε-ODEMO does well in convergence. It has the best convergence value in all test cases. On ZDT6 and DTLZ1, NSGA-II and SPEA2 are inclined to local Pareto front. NSGA-II uses crowd distance to sort populations, while INSGA-II is based on a loop crowd distance concept which improves the convergence compared to NSGA-II. DEMO performs badly in convergence, but its convergence improves obviously after adopting εdominance method. Nevertheless, its performance in convergence is far from that of Paε-ODEMO in all cases. The experimental results show that the algorithm can approach the true Pareto front well.

Seen from Table **2**, the algorithm also achieves good result in diversity. On ZDT1, it ranks only second only to INSGA-II and on DTLZ1, its result is intermediate. Apart from this, the algorithm performs best in the rest cases. In aspect of maintaining diversity, equal intervals grid is adopted in our algorithm which makes sure that the distance of any two neighbor points is fixed in a small range. As a result,

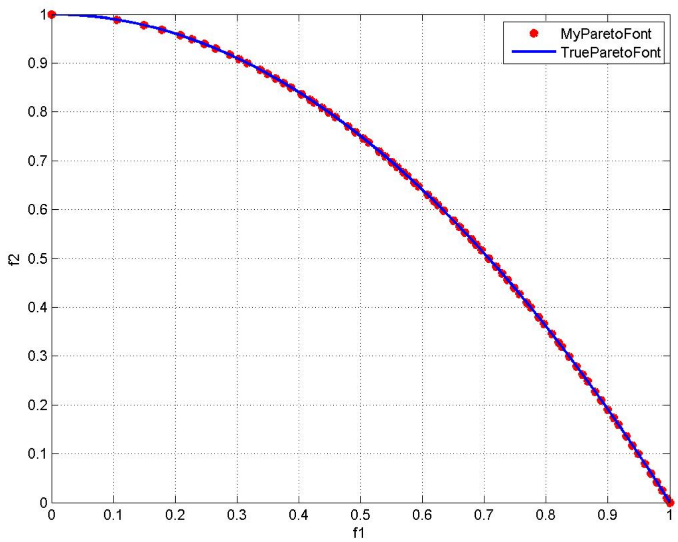

Fig. (2). Search trace to minimize 30-D ZDT2,it can get two boundary solutions easily.

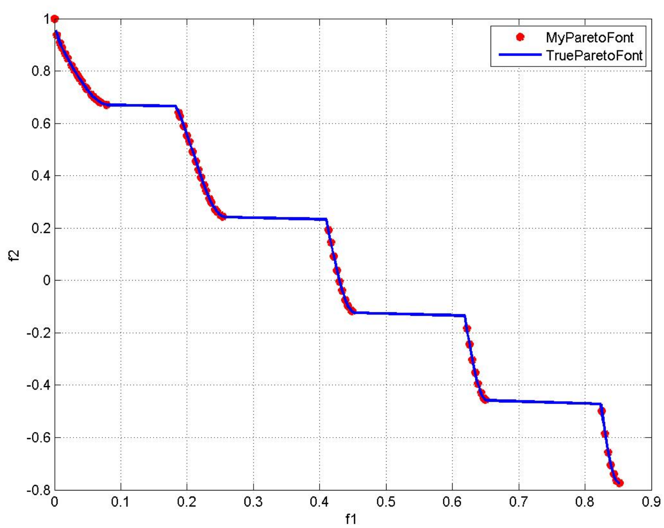

Fig. (3). Search trace to minimize 30-D ZDT3,it is discrete.

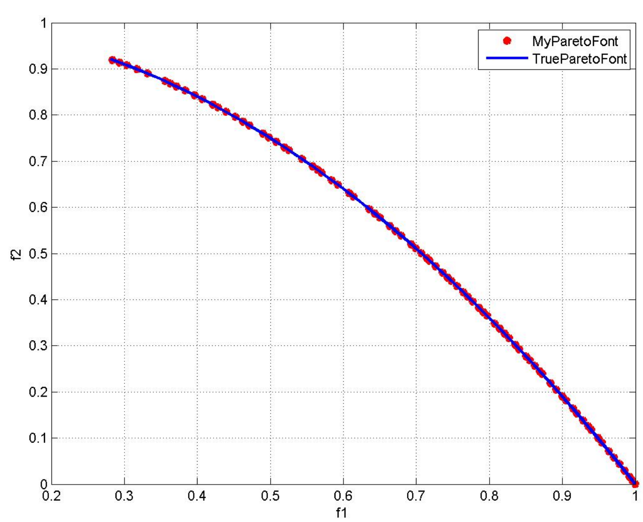

Fig. (4). Search trace to minimize 10-D ZDT6.

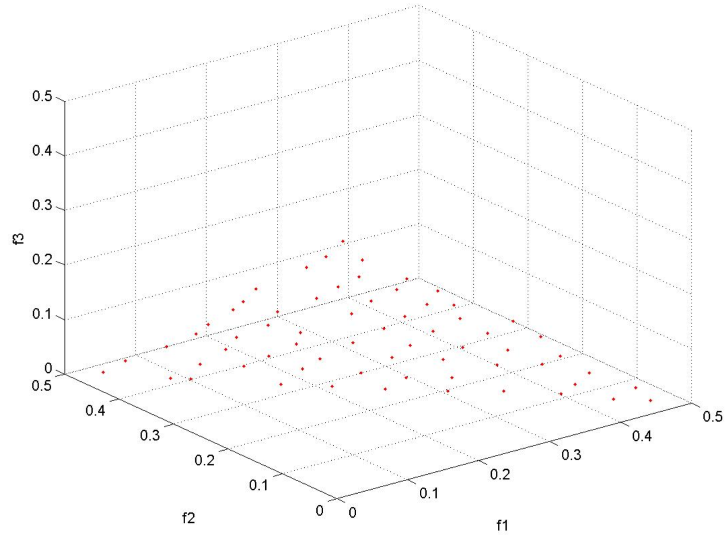

Fig. (5). Search trace to minimize 12-D DTLZ1,it is a three objective optimization problem.

all non-dominance solutions are scattered uniformly along the Pareto front. However, for some discontinuous problems, for example ZDT3, it decomposes the solution space into several parts, at this time, the equal interval grid can just make sure a good distribution in its continuous part. To this end, if the algorithm enlarges its ε value, it will result in a smaller value. Experiments show that if doubling the ε value, it can reach to 0.4. Obviously, the algorithm achieves a good result both in convergence and diversity. However, Zitzler etc. [25] point out that an algorithm's performance can't just be evaluated by numeric indicator. Therefore, in this paper, Figs. (**1-5**) of solutions run on the Matlab platform are shown to give a visual representation. Figs. (**1-5**) show the result of Paε-ODEMO run for once.

CONCLUSION

An adaptive ε dominance based orthogonal differential evolution algorithm for multi-objective optimization is proposed in this paper. The main improvement of this new approach is that it introduces an adaptive ε dominance method to update the archive population which can maintain the diversity and distribution of population in an adaptive way.

Experiments on five benchmark functions and comparison with several classical algorithm, such as NSGA-II, SPEA2, INSGA-II, AEPSO, NPCA and ε-DEMO indicate that the new algorithm Paε-ODEMO does well in both convergence and diversity. What's more, it's robust. However, seen from the figures above, it is obvious when the slop of tangent tends to 0 or 1, the points are few and scattered. Our later work will be devoted to adjusting the density of εgrid adaptively, that is to say, when the slop of tangent tends to 0 or 1, the value of ε is small which is expressed as a higher density of grid while the ε grid will be few and scattered at other parts.

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This work was supported in part by the Foundation of Henan Important Science Research Project under Grant No. 122102210488 and 132102210095, Young Core Teachers of Xuchang University.

REFERENCES

[1] W. An, "*Research on multi-objective optimization and its application*," PhD thesis of Northwestern Polytechnical University, vol. 10, 2005.

[2] J.D. Schaffer, "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms," In: *Proceedings of the 1th International Conference on Genetic Algorithms*, pp.93-100, 1986.

[3] N. Srinivas, and K. Deb, "Multi-Objective optimization using the non-dominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp.221-248, 1994.

[4] T. Murata, and H. Ishibuchi, "MOGA: Multi-objective genetic algorithms," In *Proceedings of the 2nd IEEE International Conference on Evolutionary Computing, Perth*, Australia, vol. 1, pp. 289-294, November 1995.

[5] J. Horn, and N. Nafpliotis, "*Multi-objective Optimization Using the Niched Pareto Genetic Algorithm*," Technical Report 93005, University of Illinois at Urbana Champaign, Urbana, Illinois, USA, 1993.

[6] J.D. Knowles, and D.W. Corne, "The Pareto Archive Evolutionary Strategy: A new Baseline Algorithm for Multi-objective Optimization," In: *1999 Congress on Evolutionary Computation*. Washington DC, pp. 98-105, July 1999.

[7] E. Zitzler, and L. Thiele, "*An Evolutionary Algorithm for Multi-objective Optimization: The Strength Pareto Approach*," Technical Report 43, Computer Engineering and Communication Networks Lab(TIK), Swiss Federal Institute of ethnology (ETH) Zurich, Switzer-land, May 1998.

[8] K. Deb, S. Agrawal, A. Pratab and T. Meyarivan, "*A Fast Eli-tist Non-Dominate Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II*," KanGAL report 200001, Indian Institute of Technology, Kanpur, India, vol.1917, pp. 849-858, 2000.

[9] E. Zitzler, M. Laumanns, and L. Thiele, "*SPEA2: Improving the Strength Pare-to Evolutionary Algorithm*," Technical Report TIK-Report TIK-Report 103, Swiss Federal Institute of Technology Zurich (ETH), May 2001.

[10] Y. Wang, Y. Jiao, and F. Zhang, "Uniform and orthogonal genetic algorithm for multi-objective optimization," *Journal of System Engineering*, vol. 18, no. 6, pp. 13481-486, 2003.

[11] S. Zeng, W. Wei, L. Kang, and S. Yao, "A multi-objective evolutionary algorithm based on orthogonal design," *Chinese Journal of Computers*, vol. 28, no. 7, pp. 1153-1162, 2005.

[12] S. Guan, J. Kou, and M. Li, "A co-evolutionary algorithm based on ε-constraint and augmented Lagrangian methods," *Journal of Systems Engineering and Electronics*, vol. 24, no. 9, pp. 33-37, 2002.

[13] C. Shi, Q. Li, and Z. Shi, "A quick multi-objective Evolutionary algorithm based on domination tree," *Journal of Software*, (in Chinese), vol. 18, no. 3, pp. 505-516, 2007.

[14] W. Gong, Z. Cai, and C. Ling, "ODE: A fast and robust differential evolution based on Orthogonal", //LNAI 4304: PROC of advances in Artificial Intelligence. Berlin: Springer, pp. 709-718, 2006.

[15] M. Gong, L. Jiao, D. Yang, W. Ma, "Research on Evolutionary Multi- Objective Optimization Algorithm," *Journal of Software*, vol. 2, no. 20, pp. 271-289, 2009.

[16] W. Gong, and Z. Cai, "Research on an ε-Domination Based Orthogonal Differential Algorithm for Multi-Objective Optimization," *Journal of Computer Research and Development*, vol. 4, pp. 655-666, 2009.

[17] Y. Leung, and Y. Wang, "An Orthogonal genetic algorithm with quantization for global numerical optimization," *IEEE Transaction on Evolutionary Computation*, vol. 5, no. 1, pp. 41-53, 2001.

[18] W. Gong, and Z. Cai, "An improved multi-objective differential evolution based on Pareto-adaptive ε-dominance and orthogonal design," *European Journal of Operational Research*, vol. 198, pp. 576-601, 2009.

[19] C.Y. Luo, M.Y. Chen, and C.Y. Zhang, "Improved NSGA-II algorithm with circular sorting," *Control and Decision*, vol. 25, no. 2, pp. 227-231, 2010.

[20] M.Y. Chen, C.Y. Zhang, and C.Y. Luo, "Adaptive evolutionary multi-objective particle swarm optimization algorithm," *Control and Decision*, vol. 24, no.12, pp. 1851-05, 2009.

[21] Q. He, G. Cheng, J. An, G. Dai, and L. Peng, "Pareto-based Multi-object Clonal Evolutionary Algorithm," *Computer Science*, vol. 39, no. 6A, 2012.

[22] S. Yang, S. Wang, and J. Tao, "Multi-objective optimization method based on hybrid swarm intelligence algorithm," *Computer Simulation*, vol. 29, no. 6, 2012.

[23] T. Robic, and B. Filipic, "DEMO: Differential evolution for multi-objective optimization,"//LNCS 3410:Proc. of EMO'05. Berlin: Springer, vol. 243, no. 2, pp. 520-533, 2005.

[24] K. Deb, M. Mohan, and S. Mishra, "Evaluating the domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions," *Evolutionary Computation*, vol. 13, no. 4, pp. 501-525, 2005.

[25] E. Zitzler, L. Thiele, M. Laumanns, M.C. Fonseca, and, V.G.D. Fonseca, "Performance assessment of multi-objective optimizer: An analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117-132, 2003.

Received: September 16, 2014 Revised: December 23, 2014 Accepted: December 31, 2014