

# A Multi-objective Evolutionary Algorithm of Principal Curve Model Based on Clustering Analysis

Qiong Yuan<sup>1,2</sup>, Guangming Dai<sup>1,2\*</sup>

<sup>1</sup> School of Computer Science, China University of Geosciences, Wuhan 430074, China.

<sup>2</sup> Hubei Key Laboratory of Intelligent Geo-Information Processing, China University of Geosciences, Wuhan 430074, China.

\* Corresponding author. Tel.: 13971106905; email: yqszb@126.com

Manuscript submitted March 25, 2016; accepted July 22, 2016.

doi: 10.17706/jsw.11.8.733-744

---

**Abstract:** According to the traditional GA and EDA weakness, on the basis of MMEA, the orthogonal design initialization, convergence criterion and K-means clustering analysis method were introduced in this paper and it proposed a new model multi-objective evolutionary algorithm OMEA. The practice results showed that the OMEA had been greatly improved on both convergence and diversity of the solutions, reaching a good balance on diversity and convergence. Its comprehensive performance was better than the SPEA2, NSGA-II and other traditional multi-objective evolutionary algorithm.

**Key words:** Multi-objective evolutionary algorithms, orthogonal design initialization, convergence criterion, principal curve model, K-means clustering.

---

## 1. Introduction

In solving multi-objective problems, the early GA (short for Genetic Algorithm), such as MOGA [1] proposed by Fonseca and Fleming, NSGA and NSGA-II [2], [3] proposed by Srinivas and Deb, NPGA [4] proposed by Horn and Nafpliotis, SPEA-1,2 [5], [6] proposed by Zitzler E, Thiele and so on, all had the best potential to reach the solutions to the optimal solution. But given the great blindness of its global search, it was necessary to use some special methods to make the algorithm search have a certain "purpose". EDA (Estimation of Distribution Algorithm) that was introduced to solve MOPs had achieved good results [7]. It is thought that in the process of the evolution algorithm, the distribution of the population would have certain rules. Through the establishment of a probability model on the distribution, the search could be guided to obtain more information. M. Laumanns and N. Khan et al. proposed a multi-objective optimization algorithm based on Bayes [8], [9]. Thierens and Bosman et al. proposed a multi-objective optimization algorithm based on EDA probability model [10]-[14]. Later, Tatsuya Okabe and Zhou Aimin et al. respectively proposed a multi-objective optimization algorithm based on the distribution rule model of Pareto solution set [15]-[17].

Although Zhou's MMEA [16], [17], a multi-objective algorithm based on model, had a good effect in solving multi-objective problems. Owing to using principal component analysis (PCA) as the clustering analysis algorithm, it would consume more time in each generation. At the same time, MMEA often used random initialization, so the distribution of the initial population in the early stage was chaos.

The algorithm OMEA proposed in this paper was aimed at the limitations of the above. It has the following characteristics: 1) The initial population was generated by the orthogonal design, so that the

individual could be more representative of the feasible solution region. 2) The idea of probability model was introduced into, and the ability of the algorithm to explore the unknown region was improved by establishing the model of Pareto set distribution rule. 3) K-means cluster analysis was employed instead of L-PCA clustering analysis on the traditional evolution algorithm (MMEA) based on the model in order to improve the efficiency of the algorithm.

In this paper, some numerical experiments were carried out by some of the current typical and representative test functions. The results showed that the new algorithm OMEA both in terms of the diversity and the convergence of solution had been improved. It reached a good balance between diversity and convergence, avoiding the premature convergence to the local Pareto solution set.

## 2. Orthogonal Design Initialization

If the individual was well distributed within the feasible solution domain in the population initialization, the algorithm in the next iteration process would be easier to make the individual fall in the Pareto optimal solution set. However, in the practical multi-objective optimization problems, there was no prior knowledge about the Pareto Solution Set. Quantization technology could be introduced to initialize the population in the orthogonal experiment which made the individual more representative distributed in the feasible region.

For the decision vector  $x = (x_1, x_2, \dots, x_N) \in X$ .  $x_i$  is noted as the  $i$ th factor in the orthogonal experiment.  $N$  is the number of the experimental factors. The  $N$  factors are distributed in a continuous domain which is not directly solved by traditional orthogonal design methods, so it is needed to carry out numerical quantification on the  $N$  factors. Each factor is quantified into finite numeric. Supposing  $x_i \in [l_i, u_i]$ . If it is quantified into  $Q$  levels ( $Q$  is an odd number) which are noted as  $a_{i,1}, a_{i,2}, \dots, a_{i,Q}$ , after quantization, each variable has  $Q$  levels. Therefore, the feasible solution space contains  $Q^N$  sample points. Then the orthogonal design is introduced to get a smaller and representative sample which will make the distribution regular in the feasible solution space.

To facilitate the calculation, the paper used a special kind of orthogonal array  $L_M(Q^P)$ . Here  $Q$  was a prime parameter and satisfied  $Q \geq N-1$ ,  $M = Q^2$ ,  $P = Q+1$ . Literature[18] presented the algorithm to create the orthogonal array  $L_M(Q^P)$ . It first created a basic column and then created a non-basic column. The details on it were as follows:

Algorithm 1: The generation of orthogonal population.

Structure  $L_M(Q^P)$ , where  $P = Q+1$ .

---

For (int  $i = 1; i \leq Q^2; i++$ )       $a_{i,1} = \lfloor (i-1)/Q \rfloor \bmod Q$ ;

For (int  $i = 1; i \leq Q^2; i++$ )       $a_{i,2} = (i-1) \bmod Q$ ;

For (int  $t = 1; t \leq Q-1; t++$ )       $a_{2+t} = (a_1 \times t + a_2) \bmod Q$ ;

For (int  $i = 1; i \leq M; i++$ )

    For (int  $j = 1; j \leq P; j++$ )

$a_{i,j}++$ ;

---

If the number of factors was  $N$ ,  $N$  columns from the orthogonal array  $L_M(Q^P)$  ( $P > N$ ) above were selected to get orthogonal array  $L_M(Q^P)$ .

And then,  $M$  sample points could be got from  $Q^N$  combinations. Each line represented a combination, i.e. an individual. In addition, the value of  $P$  was usually far greater than the population size  $N$  in the evolutionary algorithm. So in the algorithm initialization, the orthogonal array was first got. And then the rank sorting mechanism of NSGA-II was reused on  $P$  combination, i.e. individuals to help selecting operation. Finally, the front  $N$  individuals as the initial population from the orthogonal array were generated.

### 3. Clustering Analysis of Population

The original ideas of algorithm were owing to the potential distribution rule of Pareto solution set, which was mostly a principal curve or surface [19]. The principal curve was a one-dimensional smooth curve through the sample set as shown in Chart 1 below. In the same way, the principal surface was a two-dimensional smooth surface through the sample set.

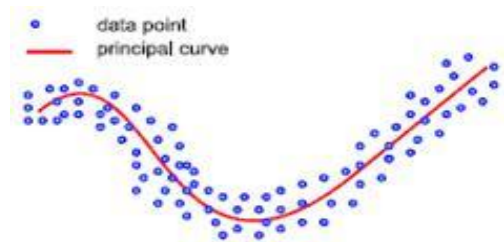


Chart 1. Principal curve diagram.

According to the existence of the potential rules, if the distribution rules of principal curve or surface could be dug out, they could guide the search for Pareto optimal solution set through the curve or surface model.

Definition 1: The population  $P(t)$  is divided into  $K$  sub populations according to the similarity between individuals.

$$C^K = \{X_i^k \mid i=1, \dots, N^k\}, k=1, \dots, K \quad (3-1)$$

$N^k$  is the number of individuals in  $k$ th sub population. And the formula satisfies:

- (1)  $C_i \neq \Phi, i=1, 2, \dots, K$ ;
- (2)  $C_i \cap C_j = \Phi, j=1, 2, \dots, K; i \neq j$ ;
- (3)  $\bigcup_{i=1}^K C_i = \{X_1, X_2, \dots, X_n\}$ .

Because there was no prior knowledge about the global distribution of the principal curve, the clustering analysis of population as shown in formula (3-1) was introduced to divide and rule the population. The distribution of individuals within each cluster was dug out. And then the probability model was established based on clustering analysis for principal curve or surface. Finally the probability model was employed to generate new offspring individuals.

#### 3.1. Principle of Principal Component Analysis (PCA) [20]

Definition 2: Let  $x = (x_1, x_2, \dots, x_n)'$  be a  $N$  dimensional random vector, and its  $i$ th principal component is:

$$F_i = u_{1i}x_1 + u_{2i}x_2 + \cdots + u_{ni}x_n = u_i'x \quad (3-2)$$

Here  $u_i'u_i = 1$ , and satisfies the following conditions:

(1)  $F_1$  is the maximum variance of all the formula like  $F = u'x$ .

(2)  $F_2$  which is not related to  $F_1$  is the maximum variance of all the formula like  $F = u'x$ .

(3)  $F_k$  ( $k = 3, 4, \dots, n$ ) which is not related to  $F_i$  ( $i \leq k-1$ ) is the maximum variance of all the formula like  $F = u'x$ .

### 3.2. Local PCA Clustering Analysis [17]

Definition 3:  $L$  is noted as a line passing through the point  $\bar{x}$ , and its direction is  $eig_1$  corresponding to the 1st biggest eigenvalue. Then the Euclidean distance between the sample point  $x_j$ ,  $j = 1, \dots, m$  ( $m$  is the total number of sample) to  $L$  is noted as  $d(x^j, L)$ . And the dividing evaluation standard is made as following:

$$\min \sum_{k=1}^K \sum_{i=1}^{N_k} [d(x^{k,i}, L_k)]^2 \quad (3-3)$$

Here  $L_k$  is the line passing through the center point  $\bar{x}^k$  of clustering  $C_k$ , and  $x^{k,i}$  is the  $i$ th sample point of clustering  $C_k$ .  $N_k$  is the total number of sample in clustering  $C_k$ . According to the dividing standard above, sample points could roughly distribute along with the clustering center line, and it could better describe the distribution rule of individuals. As shown in chart 2, the clustering shape got by the standard is roughly rectangular.

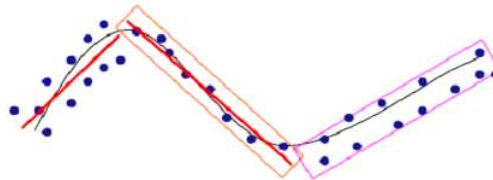


Chart 2. The result of PCA clustering.

### 3.3. K-means Clustering Analysis [21]

K-means clustering algorithm steps are as following:

(1) giving clustering number  $K$ .

(2) Randomly selecting  $K$  points  $c_1, c_2, \dots, c_K$  from  $X_1, X_2, \dots, X_n$  as  $K$  center points of clustering sets.

(3) Taking  $c_1, c_2, \dots, c_K$  as the center points to divide  $X_1, X_2, \dots, X_n$ .

If  $d(X_i, c_j) < d(X_i, c_m)$ ,  $j, m = 1, 2, \dots, K$  and  $j \neq m$ , put  $X_i$  into  $C_j$ .

(4) Calculating new center point  $c_1^*, c_2^*, \dots, c_K^*$  according to the points in  $C_1, C_2, \dots, C_K$ .

$$c_i^* = \frac{1}{|C_i|} \sum_{X_j \in C_i} X_j, i = 1, 2, \dots, K \quad (3-4)$$

Here  $|C_i|$  is the point number of  $C_i$ .

(5) If  $c_i^* = c_i, i = 1, 2, \dots, K$  ends the calculation, the current center point is the result of clustering, else let  $c_i = c_i^*$  and return (3).

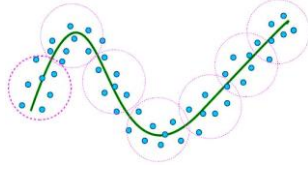


Chart 3. The result of K-means clustering analysis.

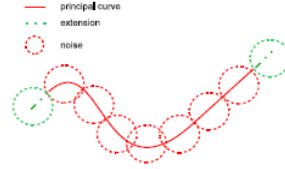


Chart 4. Introducing extended factor.

The K-means algorithm searched the result of clustering by iteration. It can simply rapidly and efficiently complete population clustering analysis. Chart 3 is the result of K-means Clustering Analysis. The clustering shape got by this standard is roughly circular.

The literature [22] used local PCA to do clustering analysis, although it can more accurately describe the distribution rule of solution set. Its computational complexity is very expensive. When evaluating each cluster, it needs to use the Jacobi [23] to calculate the eigenvalue and eigenvector of the clustering sample. The Jacobi for high dimensional matrix iterative computing process is quite time-consuming. This makes the efficiency and efficacy of local PCA algorithm not balanced. Therefore, this article uses the general k-means dividing population to improve the efficiency.

#### 4. The Establishment of Probability Model and Individual Reproduction

After clustering, it is now to establish 1-D or 2-D linear model for principal curves or surface in each  $C^k$ . It is different from traditional EDA. The probability model of evolutionary algorithm based on model is made of two parts: the deterministic model and the stochastic model. The purpose of establishing deterministic model is to obtain the rule of population distribution, and the purpose of the stochastic model is to describe the dynamic disturbance of the individual near the principal curve or surface.

In  $C^k$ , assuming  $\lambda_i^k$  as its  $i$ th biggest eigenvalue and writing its corresponding eigenvector as  $V_i^k$ , and noting the center sample point of  $C^k$  as  $\bar{X}^k, k = 1, 2, \dots, K$ , the projection of all the points in  $C^k$  on the 1st and 2nd eigenvector is calculated as following:

$$s_{1,i}^k = (X_i^k - \bar{X}^k)^T V_1^k, \quad s_{2,i}^k = (X_i^k - \bar{X}^k)^T V_2^k \quad (4-1)$$

Here  $k = 1, 2, \dots, K; i = 1, 2, \dots, N^k$ .

If there were two objective functions in MOPs, establishing 1-D linear model in each sub population, making the linear model pass through the center point  $\bar{X}^k$  and its direction being  $V_1^k$ , its model equation was as following:

$$H^k(s) = sV_1^k + \bar{X}^k \quad (4-2)$$

Here  $s_{\min}^k = \min_{i=1, \dots, N^k} \{s_{1,i}^k\}, s_{\max}^k = \max_{i=1, \dots, N^k} \{s_{1,i}^k\}$ .

If there were more than two objective functions in MOPs, establishing 2-D linear model in each sub population, the method was as following:

$$H^k(s) = s_1 V_1^k + s_2 V_2^k + \bar{X}^k \quad (4-3)$$

Here  $s = (s_1, s_2)^T$ ,  $s_1 \in [s_{1,\min}^k, s_{1,\max}^k]$ ,  $s_{1,\min}^k = \min_{i=1, \dots, N^k} \{s_{1,i}^k\}$ ,  $s_{1,\max}^k = \max_{i=1, \dots, N^k} \{s_{1,i}^k\}$ ,  $s_2 \in [s_{2,\min}^k, s_{2,\max}^k]$ ,  $s_{2,\min}^k = \min_{i=1, \dots, N^k} \{s_{2,i}^k\}$ ,  $s_{2,\max}^k = \max_{i=1, \dots, N^k} \{s_{2,i}^k\}$ .

The above formula is called as a deterministic model.

In the experiment, through the introduction of the Pareto solution set distribution rule and the Gauss model to establish the probability model of the dynamic population:

$$H^k = H^k(s) + \xi^k, \quad s \in [s_{\min}^k, s_{\max}^k] \quad (4-4)$$

Here  $H^k(s)$  is a deterministic model to describe the current feasible solution set distribution rule, while  $\xi^k$  is a random vector satisfying  $N(0, (\delta^k)^2 I)$  normal distribution.  $I$  is  $n \times n$  unit matrix.

$$\delta^k = \frac{\sum_{i=1}^{N^k} d_i^k}{N^k \sqrt{n}} \quad (4-5)$$

Here  $d_i^k$  is the Euclidean distance between the  $i$ th sample point to its principal curves in the  $k$ th sub population.

According to the probability model in formula (4-4),  $N^k$  new sample points in each sub population were generated. These points roughly distributed on the principal curves as shown in chart 4. In order to make the algorithm have better ability to search unknown fields at both ends of the principal curve, extension factor  $\varepsilon \in [0, 1]$  was introduced to extend the  $s$  in  $H^k(s)$  and  $s \in [s_{\min}^k - \varepsilon(s_{\max}^k - s_{\min}^k), s_{\max}^k + \varepsilon(s_{\max}^k - s_{\min}^k)]$ .

As the same to EDA, if the probability model was used to produce offspring in the early stage, it would make the algorithm easily premature convergence. This paper introduced the convergence criterion to determine when to perform local search:

$$\Psi(k) = \begin{cases} \sqrt{\frac{\lambda_2^k}{\lambda_1^k}} & (2-objective \text{ problems}) \\ \sqrt{\frac{\lambda_3^k}{\lambda_2^k}} & (3-objective \text{ problems}) \end{cases} \quad (4-6)$$

When  $\Psi(k) = 0$ , it indicated that the all sample points of  $k$ th sub population had distributed on the principal curves or surface. In the experiment it was set as 0.4.

If the current clustering satisfied the convergence criterion, it indicated that the current sub population points had roughly distributed near the principal curves. Right now the distribution rule of Pareto solution set had been clear, and the probability model could be employed to guide the local search in each sub population.

## 5. The Algorithm Flow and Numerical Experiments

The overall flow of OMEA which is put forward in this paper is shown in chart 5:

Genetic operation of the algorithm is in accordance with the NSGA-II [3], the smaller rank individuals are

the first to be selected. If the two individuals were in the same rank, it was the first to select the one with bigger crowding distance. This can not only make the non dominated frontier near to Pareto frontier, but also make the distribution and diversity of the solution set better.

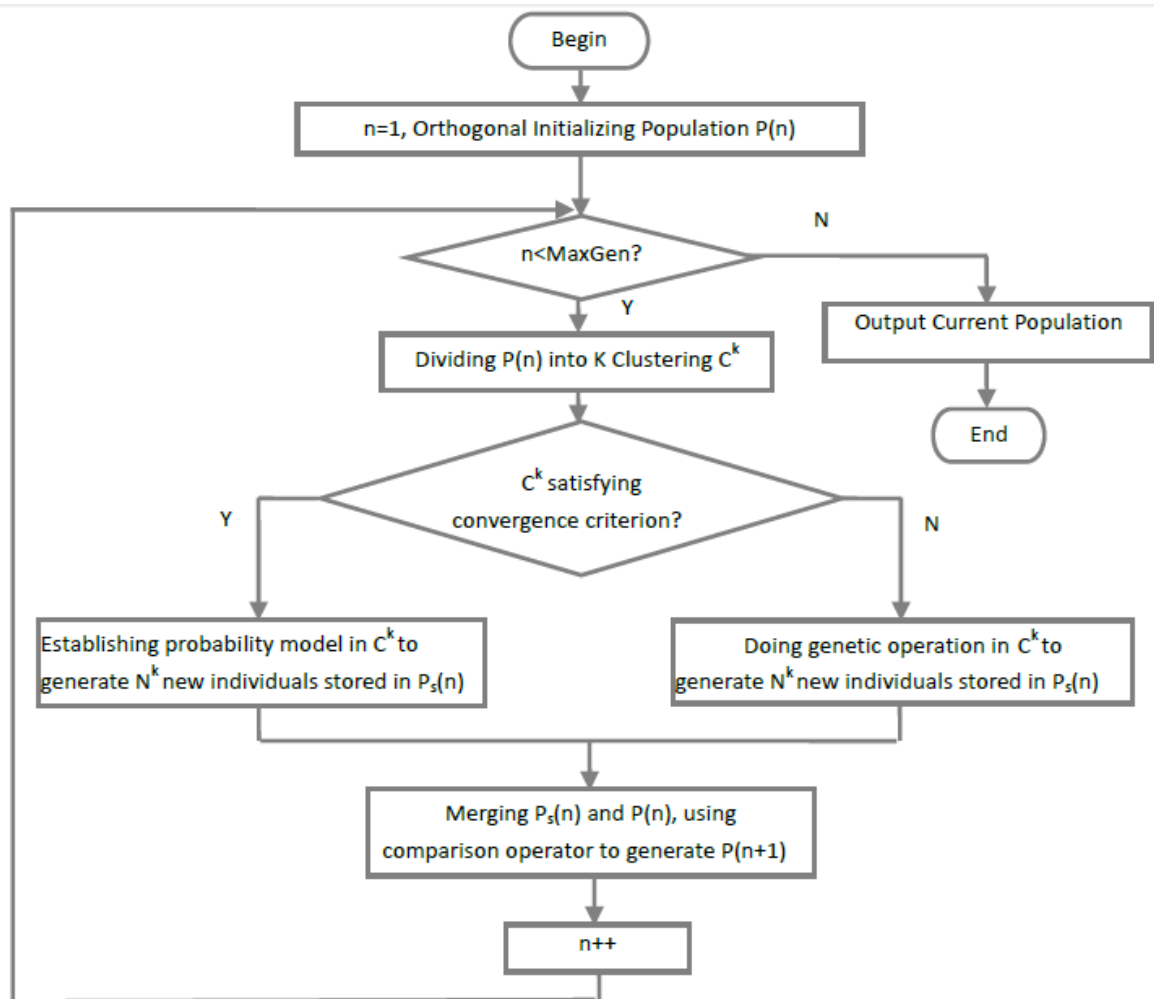


Chart 5. OMEA flow.

### 5.1. Test Functions

For the two objective problems, the OMEA selected 5 test functions of ZDT except ZDT5 to be tested. At the same time, due to the comparison with MEA/HA and MEA/HB in Zhou literature [16], [17], it improved ZDT3 to ZDT3.2[17]. The Pareto solution set of ZDT3 was  $\Omega_{ZDT3} = \{0 \leq x_1 \leq 1, x_2 = \dots = x_n = 0\}$ , while the Pareto solution set of ZDT3.2 was  $\Omega_{ZDT3.2} = \{0 \leq x_1 \leq 1, x_2 = \dots = x_n = \sqrt{x_1}\}$ , its Pareto solution set distribution became more complex. For the three objective optimization problems, it selected DTLZ1 and DTLZ7 [24] to be tested. The description on the test problems were shown as Table 1.

Table 1. Test Functions (n is the size of the decision space )

Problem	$n$	Property
ZDT1	30	High dimension, Pareto frontier is convex.
ZDT2	30	High dimension, Pareto frontier is non convex.
ZDT3	30	High dimension, Pareto frontier is discontinuous.
ZDT3.2	30	High dimension, Pareto frontier is discontinuous.

ZDT4	10	As many as 219 Pareto frontier, the closer to the true Pareto frontier is, the more densely populated the local optimal frontier is, this function is usually used to test the algorithm with ability of processing multi-modal problems.
ZDT6	10	Its distribution on the global Pareto optimal frontier is uneven, the closer to the Pareto optimal frontier is, the lower the density of solution is.
DTLZ1	7	as many as (115-1) Pareto frontier
DTLZ7	22	High dimension, as many as 22 discontinuous area on Pareto optimal solution

## 5.2. Performance Evaluation Standard

$\gamma$  metric and  $\Delta$  metric[3] put forward by Deb were widely and commonly used as evaluation standard, so the paper selected these two metrics to evaluate the algorithms. They were defined as following:

### 5.2.1. Convergence metric $\gamma$

The precondition of this standard is multi-objective Pareto optimal solution set. First of all,  $H$  points on the Pareto frontier in the objective space are sampled. Then the minimum Euclidean distance between each approximate Pareto optimal solution to the sample points is calculated, and  $\Upsilon$  is the mean of these minimum Euclidean distances. Its formula is as following:

$$\Upsilon = \sum_{i=0}^H \min d(f, f_{i,front}) / n \quad (5-1)$$

Here  $n$  is the number of solutions and  $f$  is the objective vector. The smaller the  $\gamma$  is, the better the convergence is.

### 5.2.2. Diversity metrics $\Delta$

Multi-objective algorithm was hoping to get a solution set distributed evenly on the Pareto frontier. So firstly, the Euclidean distance  $d_i$  between the two adjacent solutions and the extreme solution  $d_f, d_l$  in the objective space were calculated. And then the mean  $\bar{d}$  of these distances was calculated. Finally, the diversity metrics  $\Delta$  was expressed as the following formula :

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} |d_i - \bar{d}|}{d_f + d_l + (N-1)\bar{d}} \quad (5-2)$$

The smaller the  $\Delta$  is, the better the diversity is.

Table 2. Parameter Setting of the OMEA

Population size ( $N$ )	Generation number	Crossover probability	Mutation probability	SBX- $\eta_c$	SBM- $\eta_m$	Cluster number ( $K$ )
100	100	1.0	0.1	20	10	3

## 5.3. The Experimental Environment and Parameter Settings

The experimental environment was as follows: CPU-PIV 3.0 GHZ, Memory-2.00GB, OS-Microsoft



Windows XP Professional version 2002 Service Pack3, Development platform - Microsoft Visual Studio, NET 2008.

Experimental parameter Settings were as follows: in OMEA, employing real code, using SBX and real polynomial mutation to do genetic operation,  $\eta_c$  and  $\eta_m$  separately indicate crossover and mutation parameters. The parameters of the OMEA were set in Table 2.

#### 5.4. Numerical Experimental Result and Analysis

All the test functions were independently ran 10 times. The statistical results were as follows: 1) the mean and its standard variance of convergence metric  $\gamma$ ; 2) the mean and its standard variance of diversity metric  $\Delta$ .

(1) For the two objective test functions, the mean and the standard variance of  $\gamma$  and  $\Delta$  were shown in table 3. The results of OMEA algorithm proposed in this paper were shown in bold. The parameter settings and experimental results of other algorithms(SPEA2, NSGA-II, PAES) could be obtained from literature[3].

Table 3. The Statistical Results of the Two Objective Test Functions

Algorithm	ZDT1	
	Convergence $\gamma$	Diversity $\Delta$
NSAGA-II	0.033482 $\pm$ 0.004750	0.390307 $\pm$ 0.001876
SPEA2	0.001799 $\pm$ 0.000001	0.784525 $\pm$ 0.004440
PAES	0.082085 $\pm$ 0.008679	1.229794 $\pm$ 0.004839
<b>OMEA</b>	<b>0.00122027 <math>\pm</math> 0.000143742</b>	<b>0.352306 <math>\pm</math> 0.0272603</b>

Algorithm	ZDT2	
	Convergence $\gamma$	Diversity $\Delta$
NSAGA-II	0.072391 $\pm$ 0.031689	0.430776 $\pm$ 0.004721
SPEA2	0.001339 $\pm$ 0.000000	0.755148 $\pm$ 0.004521
PAES	0.126276 $\pm$ 0.036877	1.165942 $\pm$ 0.007682
<b>OMEA</b>	<b>0.000813525 <math>\pm</math> 0.000077</b>	<b>0.353621 <math>\pm</math> 0.0239713</b>

Algorithm	ZDT3	
	Convergence $\gamma$	Diversity $\Delta$
NSAGA-II	0.114500 $\pm$ 0.007940	0.738540 $\pm$ 0.019706
SPEA2	0.047517 $\pm$ 0.000047	0.672938 $\pm$ 0.003587
PAES]	0.023872 $\pm$ 0.000010	0.789920 $\pm$ 0.001653
<b>OMEA</b>	<b>0.00125884 <math>\pm</math> 0.0000789053</b>	<b>0.541039 <math>\pm</math> 0.0199432</b>

Algorithm	ZDT4	
	Convergence $\gamma$	Diversity $\Delta$
NSAGA-II	0.513053 $\pm$ 0.118460	0.702612 $\pm$ 0.064648
SPEA2	7.340299 $\pm$ 6.572516	0.798463 $\pm$ 0.014616
PAES	0.854816 $\pm$ 0.527238	0.870458 $\pm$ 0.101399
<b>OMEA</b>	<b>0.000968052 <math>\pm</math> 0.000103285</b>	<b>0.376844 <math>\pm</math> 0.0177152</b>

Algorithm	ZDT6	
	Convergence $\gamma$	Diversity $\Delta$
NSAGA-II	0.296564 $\pm$ 0.013135	0.668025 $\pm$ 0.009923
SPEA2	0.221138 $\pm$ 0.000449	0.849389 $\pm$ 0.002713
PAES	0.085469 $\pm$ 0.006664	1.153052 $\pm$ 0.003916
<b>OMEA</b>	<b>0.000890605 <math>\pm</math> 0.000309398</b>	<b>0.550322 <math>\pm</math> 0.0418366</b>

Because the ZDT4 has  $2^{19}$  local optimal frontier, it was commonly used to measure the algorithm with the

ability to process multi-peak function. ZDT6 optimal frontier was convex and its distribution on the objective space was uneven. So some of the traditional evolutionary algorithm was easy to fall into local solution in solving these two functions, and the distribution was extremely uneven. But from the table 3, OMEA can converge near to the global optimal frontier and the performance of the algorithm in convergence and diversity were improved significantly.

Table 4 showed the statistical results on ZDT3.2 ran 10 times by OMEA, where MEA/HA and MEA/HB were respectively proposed in literature [16], [17]. The parameter settings and experimental results of NSGA-II and SPEA2 on ZDT3.2 can also be got in literature [17]. From table 4, the OMEA had certain ability to mine the distribution rule of Pareto set. So the algorithm had better ability to do the local search and the convergence of the algorithm was better than NSGA-II and SPEA2. At the same time, from the comparison on MEA/HA and MEA/HB, the performance of OMEA was between these two algorithms. Because the convergence criterion was not introduced in MEA/HA, this made MEA/HA randomly use probability model or genetic operation to generate new individuals. If probability model was established when the population distribution was not convergent, the quality of its individual was not high. While OMEA introduced the convergence criterion and only if the current clustering satisfied the criterion, the K-means clustering was used to establish the model in the sub population to direct the local search. This made the convergence of OMEA better than MEA/HA. In MEA/HB, it employed computationally expensive L-PCA to cluster. Its model could better characterize the distribution rule of current population and make the algorithm do better on local search. But this method was at the expense on large computing time.

Table 4. The Statistical Results on ZDT3.2 Ran 10 Times by OMEA and Others

Method	Convergence $\gamma$		Diversity $\Delta$	
	ZDT3.2		ZDT3.2	
	Mean	Std.	Mean	Std.
MEA/HA	0.00995	0.00000	<b>0.42006</b>	0.01347
MEA/HB	<b>0.00392</b>	0.00000	<b>0.34086</b>	0.00902
NSGA-II	0.00754	0.00001	0.75214	0.00332
SPEA2	0.00824	0.00009	0.94170	0.00065
OMEA	<b>0.00534</b>	0.00007	<b>0.56482</b>	0.00432

(2) For the three objective test functions, it compared the convergence  $\gamma$  on DTLZ1 and DTLZ7, where  $\varepsilon$ -MOEA and  $\varepsilon$ -DEMO were the experiment results in literature [25]. Table 5 showed the statistical results on DTLZ1 and DTLZ7 ran 10 times by OMEA. The results of OMEA algorithm proposed in this paper were shown in bold. Obviously, the  $\gamma$  of OMEA was better than the others.

Table 5. The Statistical Results on DTLZ1 and DTLZ7

Algorithm	DTLZ1	DTLZ7
	Convergence $\gamma$	Convergence $\gamma$
NSAGA-II	0.574227 $\pm$ 0.502424	0.0252602 $\pm$ 0.000731946
$\varepsilon$ -MOEA	7.871200 $\pm$ 1.868758	0.062187 $\pm$ 0.032882
$\varepsilon$ -DEMO	1.687799 $\pm$ 0.182301	0.044138 $\pm$ 0.021600
<b>OMEA</b>	<b>0.00464681 <math>\pm</math> 0.000367051</b>	<b>0.0220803 <math>\pm</math> 0.00338758</b>

## 6. Conclusion

In summary, in the process of numerical experiments, 8 different functions were used to test the performance of the OMEA and the diversity and convergence performance metric  $\Delta$  and  $\gamma$  mentioned above were respectively adopted to evaluate the algorithms. From the performance values and the final optimized

Pareto optimal solutions, it can be seen that in all of the 8 test problems, the multi-objective evolutionary algorithm OMEA proposed in this paper had been greatly improved on both convergence and diversity of the solutions, reaching a good balance on diversity and convergence. Comprehensive performance was better than the SPEA2, NSGA-II and other traditional multi-objective evolutionary algorithm.

## Acknowledgment

This paper is supported by National Natural Science Foundation of China (No.61472375, No.61103144) .

## References

- [1] Fonseca, C. M., & Fleming, P. J. (1993). Genetic algorithms for multiobjective optimization: Formulation, discussion and generation. *Proceedings of the 5th International Conference on Genetic Algorithms* (pp. 416-423).
- [2] Srinivas, N., & Deb, K. (1994). Multi-objective optimization using non-dominated sorting in genetic algorithms. *Evolutionary Computation*, 2(3), 221-248
- [3] Deb, K., Pratap, A., & Garwal, A. S. E. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182-197.
- [4] Horn, J., & Nafpliotis, N. (1993). Multi-objective optimization using the niched Pareto genetic algorithm. University of Illinois at Urbana Champaign.
- [5] Zitzler, E., & Thiele, L. (1999). Multi-objective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4), 257-271.
- [6] Zitzler, E., Laumanns, M., & Thiele L. (2001). SPEA2: Improving the strength Pareto evolutionary algorithm. *Technical Report 103, Computer Engineering and Networks Laboratory (TIK)*.
- [7] S´anchez, J. M. P., Robles, V., Larra˜naga, P., Herves, V., Rosales, F., & P´erez, M. S. (2004). GA-EDA: Hybrid evolutionary algorithm using genetic and estimation of distribution algorithms. *Proceedings of the 17th International Conference on Industrial and Engineering Applications of Artificial Intelligence & Expert Systems* (pp. 361–371).
- [8] M, Laumanns., & Ocenasek, J. (2002). Bayesian optimization algorithms for multi-objective optimization. *Parallel Problem Solving From Nature - PPSN VII, Lecture Notes in Computer Science*, 298–307.
- [9] Khan, N., Goldberg, D. E., & Pelikan, M. (2002). Multi-objective bayesian optimization algorithm. *Proceedings of the Genetic and Evolutionary Computation Conference..*
- [10] Thierens, D., & Bosman, P. A. N. (2001). Multi-objective mixture-based iterated density estimation evolutionary algorithms. *Proceedings of the Genetic and Evolutionary Computation Conference* (pp. 663–670).
- [11] Larra˜naga, P., & Lozano, J. A. (2002). Estimation of distribution algorithms. *A New Tool for Evolutionary Computation*. Kluwer Publishers.
- [12] Costa, M., & Minisci, E. (2003). MOPED: A multi-objective parzen-based estimation of distribution algorithm for continuous problems. *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization* (pp. 282– 294).
- [13] Pelikan, M., Goldberg, D. E., & Lobo, F. (1999). A survey of optimization by building and using probabilistic models. *Technical Report 99018*. University of Illinois.
- [14] S´anchez, J. M. P., Robles, V., Larra˜naga, P., Herves, V., Rosales, F., & P´erez, M. S. (2004). GA-EDA: Hybrid evolutionary algorithm using genetic and estimation of distribution algorithms. *Proceedings of the 17th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems* (pp. 361–371).

- [15] Tatsuya, O., Yaochu, J., Bernhard, S., & Markus, O. (2004). Voronoi-based estimation of distribution algorithm for multi-objective optimization.
- [16] Aimin, Z., Qingfu, Z., Yaochu, J., & Edward, T. (2005). A model-based evolutionary algorithm for bi-objective optimization.
- [17] Aimin, Z., Yaochu, J., Qingfu, Z., Bernhard, S., & Edward, T. (2006). Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion.
- [18] Yiu-Wing, L. (2001). An orthogonal genetic algorithm with quantization for global numerical optimization. *IEEE Transactions on Evolutionary Computation*.
- [19] Trevor, H., & Werner, S. (1989). Principal curves. *Journal of the American Statistical Association*, 502–516.
- [20] Lindsay, I. S. (2002). A tutorial on principal components analysis.
- [21] Nathan, S. N., Christine, D. P., Ruth, S., & Angela, Y. W. (2002). An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis and 145 Machine Intelligence*, 24(7), 881-892.
- [22] Costa, M., & Minisci, E. (2003). MOPED: A multi-objective Parzen-based estimation of distribution algorithm for continuous problems. *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization* (pp. 282– 294).
- [23] Changfa, Z. (2000). *Science and Engineering Numerical Algorithm*. M. Tsinghua University Press.
- [24] Deb, K., Thiele, L., Laumanns, M., & Zitzler, E. (2001). Scalable test problems for evolutionary multi-objective optimization.
- [25] Zhihua, C., Wenying, G., & Yongqin, H. (2007). A novel differential evolution algorithm based on  $\epsilon$ -domination and orthogonal design method for multiobjective optimization. *Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization*.



**Qiong Yuan** was born in Huber, China, in 1980. She received the M.S. degree in computer application technology in Huazhong University of Science and Technology in China in 2008. Since September 2013, she has been working for her PhD degree in China University of Geosciences. Her research interests are in the area of evolutionary algorithms and optimization.



**Guangming Dai** was born in Anhui, China, in 1964. He received his PhD degree from Huazhong University of Science and Technology, China, in 2002. He is currently a professor in geosciences, China, since 2002. His main interests are in the area of evolutionary algorithms and algorithm design and analysis.