A Contour Method in Population-based Stochastic Algorithms

Ying Lin, *Student Member, IEEE,* Jun Zhang(corresponding author), *MIEEE*, and Lu-kai Lan

***Abstract*—Inspired by the contours in topography, this paper proposes a contour method for the population-based stochastic algorithms to solve the problems with continuous variables. Relying on the existed population, the contour method explores the landscape of the fitness function in the search space, which leads to effective speculation about the positions of the potential optima. The contour method is embedded into every generation of the simple genetic algorithm (SGA) for efficiency examination. The genetic algorithm with the contour method is first realized in a two-dimensional space, where the contours in topography can be directly used. Then the proposed contour method is modified to adapt high dimensional space. Numerical optimization experiments are carried out on ten benchmark functions of two and thirty dimensions. Results show that the genetic algorithm with the contour method can outperform the SGA in both solution quality and convergence speed.**

I. Introduction

ONE sort of the most popular modern computing techniques is the population-based stochastic algorithms They utilize the interactions among individuals to improve the general quality of the population and move towards the optima as a whole. The typical instances of this kind of algorithms include the genetic algorithms (GAs), the ant colony optimizer (ACO), the particle swarm optimizer (PSO) and the estimation-of-distribution algorithms (EDAs).

GAs [1] originated from the Darwinism. It obeys the law of 'survival-of-fittest' and uses the selection, crossover and mutation operators to evolve its population. ACO [2] is inspired by the food searching behaviors of natural ants. The artificial agents in ACO deposit pheromones in the underlying environment and use the accumulated pheromones to guide their search. PSO [3] learns from behaviors of bird flocking and fish schooling. The algorithm has its particles move towards some specified individuals, such as the traditional *pbest* and *gbest* in [3], and the newly proposed adjacencies in the topology in [4]. EDAs [5] estimate the probability distribution based on the good individuals in the population, and then generate new solutions by randomly sampling from the estimated distribution. None of the prototypes of the above algorithms considers exploring the landscape of the fitness function through their populations. However, as the algorithm progresses, its population obtains more and more information about the landscape of the fitness function in the search space. If ever this information could be store and analyzed, the algorithm will benefit a lot from knowing the landscape of the fitness function. This idea has been adopted in some variations of the above population-based algorithms, where promising results are found. The following paragraph reviews some of them.

Hu *et al.* [6] proposed a continuous orthogonal ant colony (COAC) algorithm, which introduce the orthogonal method into a continuous ACO. The orthogonal method defines several typical points around the selected individual. By evaluating and analyzing the solutions at these points, the COAC can grasp the local distribution of the fitness function, and then deduce the direction of promising solutions. Bosman and Grahl [7] proposed to adjust the variance of the estimated probability distribution in EDAs according to the local landscape of the fitness function over the current search area. The landscape is divided into two types: peaks and slopes. When the population is around the peak, the variance from the maximum estimation is considered to be sufficed for exploitation. However, when the population locates on the slope, the proposed EDA scales the variance for exploration. It was reported that the proposed EDA obtains significant improvements. Posik and Franc [8] proposed to find the decision ellipsoid between the selected and the discarded individuals. Then the ellipsoid function is turned into a Gaussian distribution, which can be embedded into the evolutionary algorithms to generate new individuals. The experiment proved that this schema works well in the case that the contours of the fitness function are elliptic.

The orthogonal method explores the landscape of the fitness function by some discrete points, which cannot avoid receiving incomplete information. The other two methods view the landscape in a continuous manner, but both of them lack of the deduction about the promising direction based on the information of the landscape. Besides, the improved EDA of Bosman and Graphl only utilizes the local landscape, while the schema of Posik and France is limited to unimodal quadratic functions. The idea of exploring and benefiting from the landscape of the problem is still waiting for implementation that is more efficient.

This paper proposes to explore the landscape of the fitness function in a global way. According to the landscape information, the algorithm can speculate the positions of better solutions. The proposed method is inspired by the contours in topography. It utilizes the population to draw the contours of the fitness function. The contours can shrink the location of the potential optima into small ranges. Then the optima are approximated by the centroids of the

This work was supported by NSF of China Project No.60573066 and the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, P.R. China.

Authors are with Department of Computer Science, SUN Yat-sen University, Guangzhou, P.R.China, (Jun Zhang is the corresponding author, email:junzhang@ieee.org)

978-1-4244-1823-7/08/$25.00©2008 IEEE

corresponding contours. This contour method is first implemented on problems in two-dimensional space, where the contours in topography can be directly transplanted. Then it is modified to adapt the problems in high dimensional space. The only request of the proposed contour method is that the algorithm maintains enough solutions. Thus, the contour method can be embedded into all kinds of population-based stochastic algorithms, such as GAs, ACO, PSO, and EDAs. In this paper, the proposed contour method is embedded into the simple genetic algorithm (SGA). Experiments are carried out on ten benchmark functions. Results show that the SGA with the contour method achieves significant improvements in both solution quality and convergence speed.

The remainder of this paper is organized as follows. Section II introduces the background of the contours in topography. Section III details the implementation of the SGA with the contour method in the search space of two and more dimensions. Section IV displays the experiments and discussions. Section V draws a conclusion.

II. Contours in Topography

A. Some basic knowledge about contours

Topography is a science and technology that describes the physical features of an area [9]. One of the most useful tools in topography is the map of contours. A contour is a line drawn on a plan that connects the points of the locations with equal elevations [10]. A group of contours can show the hills, valleys, and the steepness of slopes in the specified area.

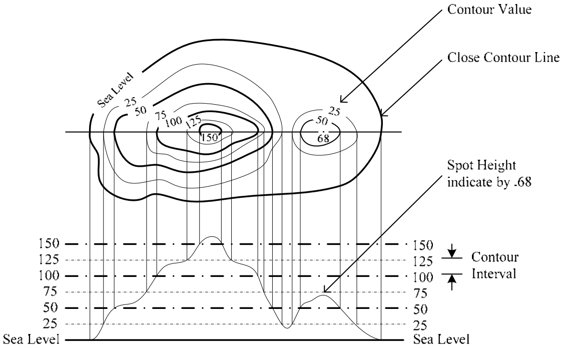


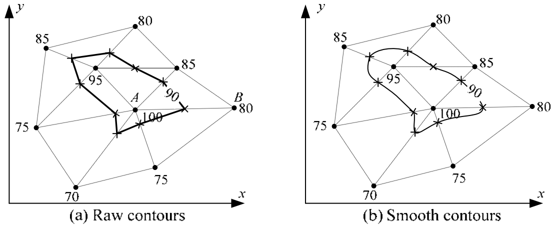
Fig.1 The profile and the contours of a landform

Fig.1 displays the profile of a landform and its corresponding contours. For better interpretations in the following sections, we also introduces some special terms for contours in this figure. As it can be seen, the dash-dot lines with equal interval indicate the elevations of the locations on the landform. The contours are drawn by connecting the points that correspond to the locations at the same height. The contour interval is the value of the difference in elevation between two contours. The elevation of each contour line is named the contour value. The height between the contours is the spot height. In Fig.1, the contour interval is 25, the contour values are {25, 50, 75, 100, 125, 150}, and the elevation of the peak of the lower hill is a spot height, indicated by '.68'.

Every contour line must be close and it only crosses the others when a cliff is detected. Thus, the summit can be circled out by the contours with a proper contour value. For example, the two summits of the landform in Fig.1 are enclosed by the contour at 50 and 150 respectively. With the idea of the contours bared in mind, the proposed contour method manages to draw the contour of the optimized objective in the population-based algorithm. Since most of the optimal values correspond to the summits of the objective functions, the contours narrow the search of the optimal solution to a small range within a specified contour value.

The construction of contours

The traditional way to generate contours stands on the basis of a uniform grid network [9][10]. However, the grid network requests large storage space and many computation efforts, which is not welcomed by its development in GA's. Thus, this paper chooses a more deliberated way to generate contours.



(a) Raw contours

(b) Smooth contours

Fig.2 A contour on the TIN

Most of the contours are generated on the basis of a triangulated irregular network (TIN) in now days [10]. In the TIN, the points whose elevations are already known are connected into contiguous and non-overlapping delaunay triangles. The construction of delaunay triangles follows certain rules [11] and promises the most accurate contours. The sides of the triangles are analyzed and the contours crossing the sides are interpolated. Fig.2 shows a contour on the TIN. The solid points and the numbers attached to them are the locations with the heights measured by surveyors. The TIN is constructed as the dotted lines in Fig.2 (a). When all the triangles are traversed, the linear interpolation on every possible side is executed orderly. In Fig.2 (a), the symbol '×' represents the interpolated point, whose position is set by (x, y) from the linear interpolation formula (1).

$$\begin{cases} x = x_A + \dfrac{h - h_A}{h_B - h_A}(x_B - x_A) \\ y = y_A + \dfrac{h - h_A}{h_B - h_A}(y_B - y_A) \end{cases}. \tag{1}$$

In (1), A and B are two points on one side of a triangle. (x_A, y_A) and (x_B, y_B) are the positions of A and B on the x-y plane. h_A and h_B are the elevations of these two points. h is the contour value, which equals 90 in Fig.2. After all the interpolated points are figured out, a polygon connects all of them, as it is shown in Fig.2 (a). In Fig.2 (b), the polygon is smoothed for pleasant view.

The construction of the TIN costs most computation when drawing contours by computer. It's also the most difficult part when importing contours into population-based algorithms, which will be illustrated in Section 3.

III. Implementation of the Contour Method in Genetic Algorithms

Relying on the solutions that already found, the contours of the optimized objective can be drawn. The contours help the algorithm to grasp the landscape of the fitness function and circle out the potential summits. Therefore, the contours lead the algorithm to converge more efficiently towards the locations of the optima.

This section utilizes the SGA as the carrier of the contour method. A flowchart of the SGA is displayed in Fig.3, but the details of the SGA are skipped. When the solution space is only two dimensional, the contour method is easy to implemented. However, if the dimension of the solution space is three or more, the contour method needs to be modified.

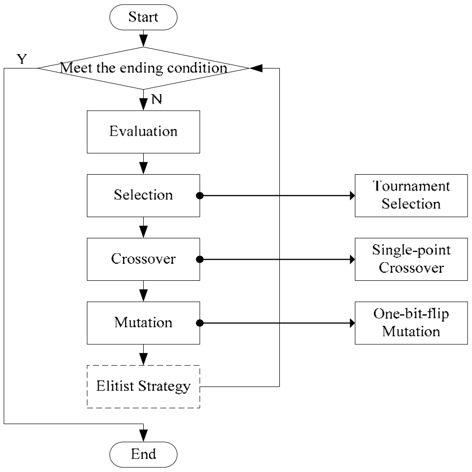


Fig.3. The flowchart of the SGA

A. *The original contour method in two-dimensional continuous problems*

For the continuous problem of two dimensions, every chromosome in SGA can be expressed as $X = (x^1, x^2; f)$. x^1 and x^2 position the chromosome in the solution space. f is the value of the fitness function in respect to x^1 and x^2.

After every generation, the unduplicated solution vectors from the current population set up the TIN in the solution space, which is shown by Fig.4 (a). The local optima, which are the chromosomes better than its adjacencies in the TIN, are found by the traversal of the triangles. As for the example in Fig.4, the two optima are denoted by the hollow spots.

The contour method circles out the potential summits by drawing enclosed contour lines around the local optima, as it's shown by Fig.4 (b). The symbol '×' represents the interpolated pointed set by (2)

$$\begin{cases} x^{1'} = x_0^1 + \dfrac{f' - f_0}{f_1 - f_0}(x_1^1 - x_0^1) \\ x^{2'} = x_0^2 + \dfrac{f' - f_0}{f_1 - f_0}(x_1^2 - x_0^2) \end{cases}, \tag{2}$$

where $(x_0^1, x_0^2; f_0)$ and $(x_1^1, x_1^2; f_1)$ are the local optima and their corresponding adjacencies, respectively. f' is the contour value set by (3),

$$f' = \frac{1}{\rho} \cdot f_0 \tag{3}$$

and ρ is a constant belonging to (0, 1). Notice that the contour lines here no longer cross the sides but the elongated sides of the triangles. This setting utilizes the existed population to speculate the location of the better solutions. f' rises with the growing f_0 during the evolutionary process of the SGA, which makes the contours get close to the summits step by step. In Fig.4 (b), f' are set as 22 and 21 respectively. The smooth operation in the contour drawing is ignored, because it does not contribute to the performance of the algorithm.

The final step of the contour method determines the summits within the contours. This paper uses the centroid of the contour line to approximate the summit. The position of the centroid is calculated by

$$x^i = \frac{\sum_{X_k \in \Gamma} x_k^i}{|\Gamma|}, i = 1,2 . \tag{4}$$

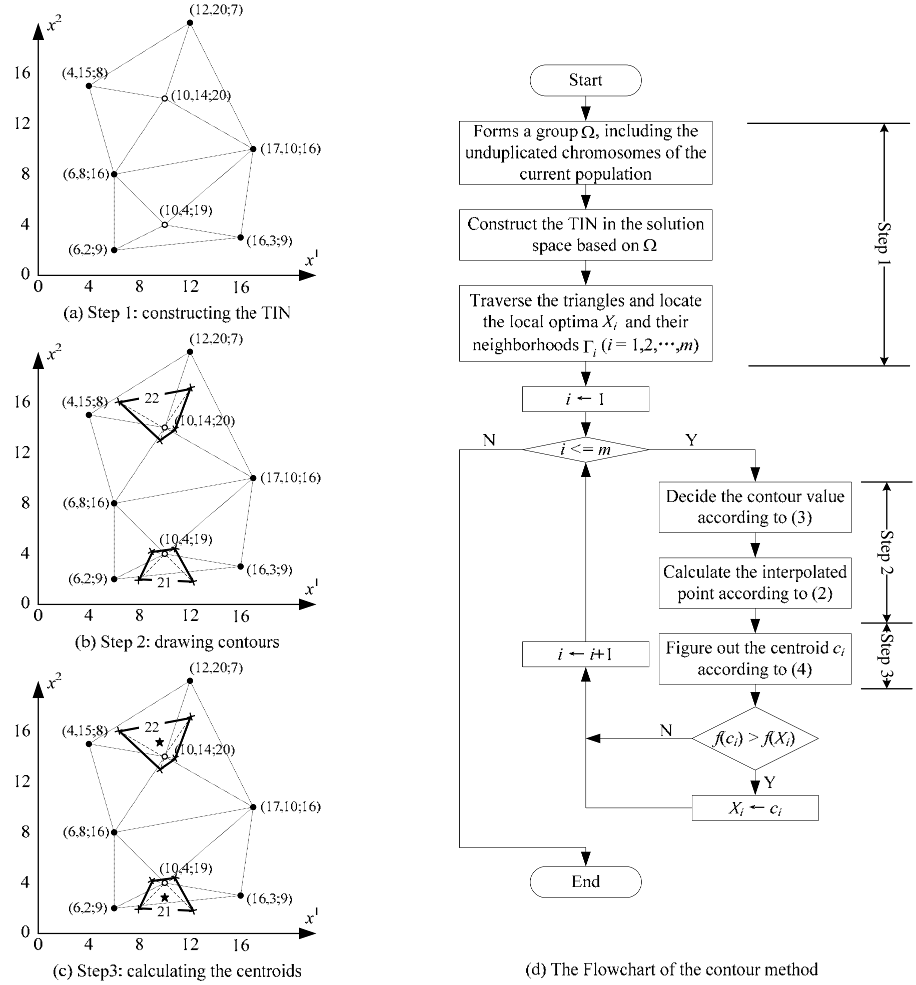
In (4), the symbol Γ is the neighborhood of one local optimum, which concludes all the chromosomes adjacent to the local optimum in the TIN. In Fig.4 (c), two pentacles denote the centroids of the two contour lines. Suppose the fitness of the centroid is better than its corresponding local optimum, the centroid replaces the local optimum in the population; otherwise, the population remains unchanged.

The flowchart in Fig.4 (d) summarizes the above steps of the contour method. This method is inserted right after the elitist strategy of the SGA.

B. *The modified contour method in high dimensional continuous problems*

The problem of the contour method in high dimensional space lies in the unfeasible construction of the TIN. Therefore, this paper replaces the TIN with a much simpler network. Moreover, in order to assure the legality of the contours based on this network, the contour method employs an extra mechanism of examination.

Notice that in the contour method introduced above, the contours are only constructed around the local optima. It is not difficult to figure out that the local optima must be the


x²
x¹
(12,20;7)
(4,15;8)
(10,14;20)
(17,10;16)
(6,8;16)
(10,4;19)
(16,3;9)
(6,2;9)
22
21
0
4
8
12
16
Start
Forms a group Ω, including the unduplicated chromosomes of the current population
Construct the TIN in the solution space based on Ω
Traverse the triangles and locate the local optima X_i and their neighborhoods Γ_i (i = 1,2,···,m)
Step 1
i ← 1
i <= m
N
Y
Decide the contour value according to (3)
Calculate the interpolated point according to (2)
Step 2
Figure out the centroid c_i according to (4)
Step 3
i ← i+1
$f(c_i) > f(X_i)$
X_i ← c_i
End

(a) Step 1: constructing the TIN

(b) Step 2: drawing contours

(c) Step3: calculating the centroids

(d) The Flowchart of the contour method

Fig.4. The process of the contour method in two dimensional continuous problem

several best chromosomes in the population. The TIN, which is not related to the local optima, is useless. The new network utilizes the above conclusion and only establishes connection between the best chromosomes and their nearest neighbors.

In the continuous problem of D dimensions, every chromosomes can be represented by $X = (x^1, x^2, \cdots, x^D)$. Firstly, the chromosomes in the current population are sorted by descending fitness. The several best chromosomes are selected to form an excellent set Φ . The size of Φ is in proportion to the size of the population, that is

$$|\Phi| = \lceil \alpha \cdot n \rceil, \quad (5)$$

where n is the total of the chromosomes and α is a predefined constant in (0,1]. The distances between the chromosome in Φ and the other chromosomes are calculated by

$$d_{ij} = \sqrt{\sum_{k=1}^{D} (x_i^k - x_j^k)^2}, i = 1,2,\cdots,|\Phi|, j = 1,2,\cdots,n \ . \quad (6)$$

For every chromosome in Φ , its neighborhood Γ_i in the new network includes the η nearest chromosomes according to (6). Following the above construction process, it's possible that several $X_i \in \Phi$ share the same neighborhood, which means

$$\Gamma_i = \Gamma_j, i \neq j \text{ and } X_i, X_j \in \Phi \ . \quad (7)$$

Therefore, the local optima in the new network do not correspond to the chromosomes in Φ one by one but are the best chromosomes in every Γ_i .

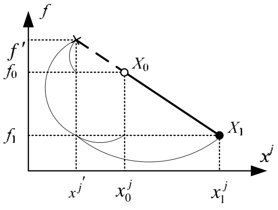


Fig.5. The interpolation on every dimension

In every Γ_i , the contour method draws a contour line around its local optimum, whose contour value f' is still set by (3). The linear interpolation is executed in every dimension as in Fig.5. Thus, with X_0 and X_1 representing the local optima and one of its adjacencies, the interpolated point

X' is positioned by (8) on every dimension.

$$x^{j'} = x_0^j + \frac{f' - f_0}{f_1 - f_0}(x_1^j - x_0^j),\ j = 1,2,\cdots,D,\ x_0^j \in X_0,\ x_1^j \in X_1 \quad (8)$$

Since the TIN has been replaced, the legality of the interpolated points can not be assured. That means the real fitness of the interpolated points may have great difference with the contour value. This phenomena will do hurt to the efficiency of the contour method, because the contours are no longer accurate. The phenomenon is shown by C_2 in Fig.6. The hollow spot *A* and *B* produces an interpolated point *X'*. However, as it can be seen, the contour value, which is considered as the fitness of *X'*, is far from the real fitness at the position of *X'*. Therefore, the contour has lost its meaning and the centroid C_2 is of bad quality. To the contrast, the centroid C_1 from the accurate contour can achieve near optimal value. In order to avoid the above problem, the contour method imports an examine mechanism. It determines to save the interpolated points or not by the difference between their fitness and the contour values. If the interpolated points that survive the examination are not enough to draw a contour, the algorithm will skip the current Γ_i. If the interpolated points left can still form a legal contour, the summits within the contours will be approximated by the centroid, whose formula can be deduced from (4).

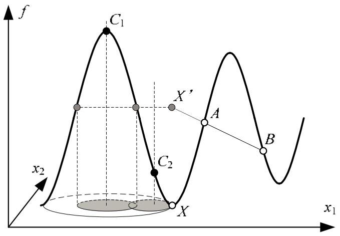


Fig.6. The accurate and inaccurate contours

IV. Experiments and discussions

In this section, contour method is testified on the function optimization problem. We will first compare the performances of the GA with original contour method (OCGA), the GA with modified contour method (MCGA), and the SGA by optimizing functions of two dimensions. Then the modified contour method is examined in high dimensional space.

The ten test functions are listed in Table 1. f1 to f6 are unimodal functions while the others are multimodal. The 'OPT' are the theoretical optima of these functions. Since all the functions are minimum problems, the fitness is the minus objective value, which is smaller than 0. Thus, equation (3) is transformed into (9),

$$f' = \rho \cdot f_0, \quad (9)$$

where ρ is set as the golden ratio, 0.618. The size of the population is 20, while the crossover and the mutation probabilities are 0.7 and 0.07 respectively. The other parameters in the contour method are set as follows: $\alpha = 0.2$, $|\Gamma_i| = 5$. When the legal interpolated points are less than three, the contour will not be drawn. The error bound is set to 0.1, which means the trial succeed when it finds a solution *X* satisfies

$$| f_i(X) - \text{OPT} | < 0.1 . \quad (10)$$

All the results come from 100 independent trials.

TABLE 1 TEST FUNCTIONS

Equation	Domain	OPT
$f_1 = \sum_{i=1}^{D} (x^i)^2$	$[-100, 100]^D$	0
$f_2 = \sum_{i=1}^{D} \lvert x^i \rvert + \prod_{i=1}^{D} \lvert x^i \rvert$	$[-10, 10]^D$	0
$f_3 = \sum_{i=1}^{D} (\sum_{j=1}^{i} x^j)^2$	$[-100, 100]^D$	0
$f_4 = \max_i(\lvert x^i \rvert), 1 \le i \le D$	$[-100, 100]^D$	0
$f_5 = \sum_{i=1}^{D} \lfloor x^i + 0.5 \rfloor^2$	$[-100, 100]^D$	0
$f_6 = \sum_{i=1}^{D} i(x^i)^4 + \text{random}[0,1)$	$[-100, 100]^D$	0
$f_7 = \sum_{i=1}^{D} [(x^i)^2 - 10\cos(2\pi x^i) + 10]$	$[-5.12, 5.12]^D$	0
$f_8 = -20\exp(-0.2\sqrt{\sum_{i=1}^{D} (x^i)^2 / D})$ $- \exp(\sum_{i=1}^{D} \cos 2\pi x^i / D) + 20 + e$	$[-32, 32]^D$	0
$f_9 = \sum_{i=1}^{D} (x^i)^2 / 4000$ $- \prod_{i=1}^{D} \cos(x^i / \sqrt{i}) + 1$	$[-500, 500]^D$	0
$f_{10} = \pi / n\{10\sin^2(3\pi y^1)$ $+ \sum_{i=1}^{D-1} (y^i - 1)^2[1 + \sin^2(3\pi y^{i+1})]$ $+ (y_D - 1)^2[1 + \sin^2(2\pi x^D)]\}$ $+ \sum_{i=1}^{D} u(x^i, 10, 100, 4)$ *	$[-50, 50]^D$	\0

$$* \ y^i = 1 + (x^i + 1)/4;\ u(x^i, a, k, m) = \begin{cases} k(x^i - a)^m & x^i > a \\ 0 & -a \le x^i \le a \\ k(-x^i - a)^m & x^i < -a \end{cases}$$

A. The comparison on two-dimensional (2D) functions

The number of the maximum evaluations spent in one trial is 100,000 on 2D functions. The program of the TIN construction used in the original contour method is free software coded in C++ by the Free Soft Foundation, Boston, Massachusetts.

Table 2 shows the comparison of the solution quality among the SGA, the OCGA, and the MCGA. The 'ok%' is the successful rate. All the advantageous values are bold in font type. As it can be seen, the contour methods improve the solution quality of the SGA dramatically. In most cases, the MCGA achieves the best results.

Table 3 displays the comparison of the convergence speed between the GAs with different contour methods. 'meanEr' is the evaluations that the algorithm uses before it succeeds. The

MCGA converges faster than the original one on all the functions. With the experiments carried out on a Dell PC of Pentium 4 CPU at 2.80 GHZ, 'avgErTime' and 'avgTime' record the average physical time to reach the successful and the final solutions respectively. These two values emphasize the difference of the runtime between the OCGA and the MCGA. The results show that the replacement of the TIN not only extends the contour method to high dimensional space, but also saves a lot of computational time.

TABLE 2 COMPARISON OF SOLUTION QUALITY ON 2-DIMEINSIOINAL FUNCTIONS

f	SGA				OCGA				MCGA			
	best	median	mean	ok%	best	median	mean	ok%	best	median	mean	ok%
f_1	1.86×10^{-5}	9.31×10^{-5}	1.15×10^{-3}	100	1.43×10^{-19}	4.39×10^{-16}	8.39×10^{-16}	100	0	2.04×10^{-208}	6.47×10^{-35}	100
f_2	6.01×10	1.83×10^{-3}	3.59×10^{-3}	100	8.47×10^{-82}	4.57×10^{-57}	3.02×10^{-50}	100	0	4.94×10^{-324}	4.94×10^{-324}	100
f_3	9.31×10^{-6}	7.54×10^{-4}	5.25×10^{-3}	100	1.91×10^{-22}	3.65×10^{-18}	6.02×10^{-17}	100	9.88×10^{-324}	6.67×10^{-164}	7.41×10^{-33}	100
f_4	3.05×10^{-3}	0.0153	0.0330	97	1.16×10^{-46}	3.50×10^{-39}	5.25×10^{-36}	100	4.94×10^{-324}	4.94×10^{-324}	9.88×10^{-324}	100
f_5	0	0	0	100	0	0	0	99	0	0	0	100
f_6	2.60×10^{-10}	1.41×10^{-8}	3.27×10^{-6}	100	1.12×10^{-7}	2.49×10^{-03}	4.55×10^{-03}	100	0	4.23×10^{-26}	7.69×10^{-23}	100
f_7	9.69×10^{-6}	4.84×10^{-5}	6.40×10^{-4}	100	1.71×10^{-11}	4.38×10^{-11}	4.46×10^{-11}	100	0	3.55×10^{-15}	6.30×10^{-13}	100
f_8	3.96×10^{-3}	0.0122	0.0294	99	5.89×10^{-16}	5.89×10^{-16}	5.89×10^{-16}	100	5.89×10^{-16}	5.89×10^{-16}	5.89×10^{-16}	100
f_9	2.52×10^{-4}	5.61×10^{-3}	9.47×10^{-3}	100	1.83×10^{-12}	7.40×10^{-03}	5.35×10^{-03}	100	1.11×10^{-16}	7.40×10^{-3}	5.18×10^{-03}	100
f_{10}	2.66×10^{-6}	6.71×10^{-5}	1.08×10^{-3}	100	7.14×10^{-17}	3.70×10^{-15}	1.21×10^{-13}	100	5.45×10^{-31}	6.79×10^{-29}	5.86×10^{-18}	100

TABLE 3 COMPARISON OF CONVERGENCE SPEED ON 2-DIMENSONAL FUNCTIONS

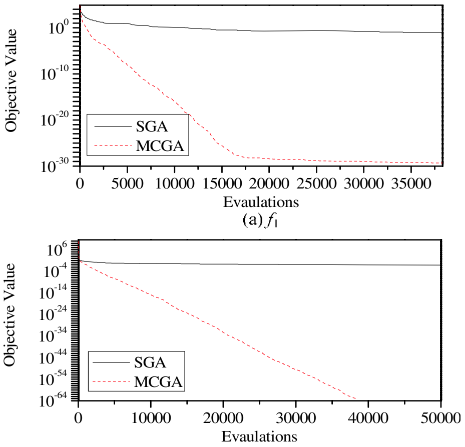
f	OCGA			MCGA		
	meanEr	avgErTime (s)	avgTime (s)	meanEr	avgErTime (s)	avgTime (s)
f_1	1976	0.06949	2.56968	582	0.00236	0.08207
f_2	494	0.02206	2.42248	225	0.00016	0.28268
f_3	2928	0.10248	2.84452	661	0.00143	0.16406
f_4	3862	0.10937	2.34984	474	0.00141	0.3646
f_5	2201	0.05132	0.033	1980	0.0103	0.0103
f_6	4774	0.14483	1.60009	896	0.00299	0.04186
f_7	1581	0.04137	1.87295	1577	0.00864	0.01598
f_8	943	0.03065	0.23123	750	0.0036	0.01343
f_9	3872	0.08095	1.4517	2722	0.01453	0.1477
f_{10}	1744	0.05775	2.5248	1613	0.01091	0.03282

From the above discussion, conclusion can be drawn that the contour method is indeed efficient. Moreover, the modified contour method is feasible and generally outperforms the original one.

B. The Comparison on thirty-dimensional functions

The number of the maximum evaluations in one trial is set to 1,000,000 when D is 30.

Table 4 lists the comparison between the MCGA and the SGA on different items. The columns of 'best', 'worst', and 'mean' correspond to the best, the worst and the average results in 100 independent trials. The meanings of 'meanEr' and 'ok%' are the same as those in Table 2 and 3. Table 4 shows that the modified contour method outperforms the SGA in every aspect. As it can be seen in Table 4, the SGA fails to find satisfying results on f_3 and f_4, while the MCGA can solve the two functions by 100% successful rate. On the other unimodal functions, the MCGA shows obvious advantages on both solution quality and convergence speed. Fig.7 picks out f_1 and f_2 as two typical instances to show the convergence of the SGA and the MCGA. The dash line and the solid line represent the SGA and the MCGA respectively. The dash line is steeper than the solid line, implying that the MCGA converges faster. Besides, the dash line ends below the solid lines, which means the MCGA achieves better results than the SGA does.



(a) f_1

(b) f_2

Fig.7. The convergence of f_1 and f_2

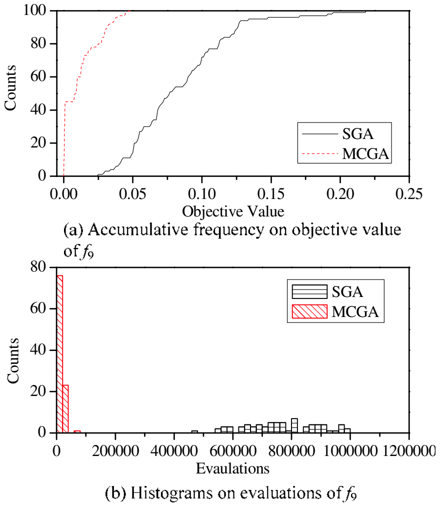
Regarding to the four multimodal functions, Table 4 shows that the MCGA still has obvious advantage on f_7, f_8 and f_{10}. As for f_9, the MCGA can find better results, yet its performance fluctuates too much to show its advantages in terms of mean value. Fig.8 uses two statistical graphs to illustrate the situation of f_9. Fig.8 (a) shows the curves of accumulative frequency [12] of the objective values from the SGA and the MCGA. The dash line, which is for the MCGA, rises sharply and reaches 100 counts before 0.05. That means most of the

results found by the MCGA concentrate on a small range of (0, 0.05). The solid line for the SGA spans from 0 to 0.22,

TABLE 4 COMPARISON ON 30-DIMENSIONAL FUNCTIONS

f	SGA					MCGA				
	best	worst	mean	meanEr	ok%	best	worst	mean	meanEr	ok%
f_1	0.00766	0.09774	0.03895	605438	100	2.27×10^{-30}	1.41×10^{-29}	6.17×10^{-30}	28471	100
f_2	0.04151	0.10987	0.07132	709156	98	1.09×10^{-67}	1.35×10^{-59}	1.40×10^{-61}	32823	100
f_3	85.8558	466.925	202.159	-	0	6.5×10^{-3}	0.09551	0.02285	699703	100
f_4	0.57070	4.59914	1.85125	-	0	4.83×10^{-16}	3.81×10^{-14}	5.51×10^{-15}	131578	100
f_5	0	0	0	106480	100	0	0	0	95481	100
f_6	9.44×10^{-4}	0.06497	6.304×10^{-3}	441999	100	1.58×10^{-19}	7.19×10^{-18}	1.05×10^{-18}	60823	100
f_7	7.31×10^{-3}	0.04482	0.01703	401390	100	$5.98 \times 1\ 0^{-11}$	1.08×10^{-10}	8.46×10^{-11}	80201	100
f_8	0.02681	0.08849	0.05238	559681	100	1.48×10^{-14}	2.19×10^{-14}	1.75×10^{-14}	26542	100
f_9	0.02494	0.21894	0.08379	766508	72	7.71×10^{-12}	0.04661	0.01121	19118	100
f_{10}	4.15×10^{-5}	2.73e-03	2.64×10^{-4}	48038	100	1.05×10^{-17}	5.69×10^{-17}	2.28×10^{-17}	35754	100

which situation of f_9. Fig.8 (a) shows the curves of accumulative frequency [12] of the objective values from the SGA and the MCGA. The dash line, which is for the MCGA, rises sharply and reaches 100 counts before 0.05. That means most of the results found by the MCGA concentrate on a small range of(0, 0.05). The solid line for the SGA spans from 0 to 0.22, which implies the objective values found by SGA distributes on a much larger range. Thus, even though the mean values of the SGA and MCGA are similar on f_9, the solution quality of the MCGA is indeed much better than the SGA. Fig.8 (b) shows the histograms [12] of the values from 'meanEr'. The histogram shaded by the diagonal lines stands for the MCGA while the other is for the SGA. Apparently, the MCGA reaches successful solutions much earlier than the SGA.



(a) Accumulative frequency on objective value of f_9

(b) Histograms on evaluations of f_9

Fig.8. Statistical graphs of f_9

Concluded from the above discussions, the MCGA outperforms the SGA on all the test functions. The advantages of the MCGA are both obvious in the precision and the speed. Therefore, it should be said that the modified contour methodis feasible and effective in high dimensional space.

V. CONCLUSIONS

Inspired by the contours in topography, this paper presents a contour method to explore the landscape of the fitness function. By drawing the contours, the algorithms can locate the ranges of the potential summits effectively. The contour method can be modified to adapt the high dimensional space. After embedded into the SGA, the contour method is testified on function optimization problem. Experimental results show that the contour method is very effective in improving the solution quality and convergence speed.

Even though this paper only uses the SGA as the carrier of the contour method, the contour method can be easily extended to the other population-based stochastic algorithms.

REFERENCES

[1] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Mich, 1975.

[2] M. Dorigo and T. Stützle, *Ant Colony Optimization*. The MIT Press, Cambridge, Massachusetts, 2004.

[3] Y. Shi and R. C. Eberhart, "A modified particle swarm optimizer," in *Proceedings of the IEEE World Congress of Computer Intelligence (CEC '98)* (Anchorage, AK, USA, May 4-9, 1998). Springer Press, Heidelberg, 1998, pp. 69-73.

[4] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: simpler, maybe better," in *IEEE Transaction on Evolutionary Computation*, vol. 8, pp. 204 – 210, Jun. 2004.

[5] H. Mühlenbein, G. Paaβ, "From recombination of genes to the estimation of distributions I," in *Binary Parameters LNCS*, vol. 1411, pp. 178 – 187. 1996.

[6] X. M. Hu, J. Zhang, and Y. Li, "Orthogonal methods based ant colony search for solving continuous optimization problems," in *Journal of Computer Science and Technology*, vol. 23, pp. 2 – 18, Jan. 2008.

[7] P. A. N. Bosman and J. Grahl, "Matching inductive search bias and problem structure in continuous Estimation-of-Distribution Algorithms," in *European Journal of Operation Research*, vol. 185, pp. 1246 – 1264, 2008.

[8] P. Posik and V. Franc, "Estimation of fitness landscape contours in EAs," in *Proceeding of GECCO 2007*, London, U.K., pp. 562 – 569, July. 2007.

[9] J. D. Bies and R. A. Long, *Mapping and Topographic Drafting*, South-Western Publishing Co. Cincinnati, Ohio. 1983.

[10] F. B. Kavanagh, *Geomatics*. Upper Saddle River, New Jersey, 2002.

[11] W. Evans, D. Kirkpatrick, and G. Townsend, “Right-triangulated irregular networks, ”. *Algorithmia*, vol. 30, pp. 264-286, Dec. 2001.

[12] J. F. Frenzel, “Genetic algorithms,” in *IEEE Potential*, vol. 12, pp. 21-24, Oct. 1993.

[13] Mann, P. S. *Introductory statistics*. 5th ed. Wiley, New Jersey. 2003.