Available online at www.sciencedirect.com
ScienceDirect
Applied Soft Computing 8 (2008) 88–95
Applied Soft Computing
www.elsevier.com/locate/asoc

Crowding clustering genetic algorithm for multimodal function optimization

Ling Qing [a,*], Wu Gang [a], Yang Zaiyue [b], Wang Qiuping [c]

[a] *Department of Automation, University of Science and Technology of China, China*
[b] *Department of Mechanical Engineering, University of Hong Kong, Hong Kong, China*
[c] *National Synchrotron Radiation Laboratory, University of Science and Technology of China, China*

Received 9 January 2006; received in revised form 29 July 2006; accepted 26 October 2006
Available online 22 December 2006

Abstract

Interest in multimodal function optimization is expanding rapidly since real-world optimization problems often require location of multiple optima in a search space. In this paper, we propose a novel genetic algorithm which combines crowding and clustering for multimodal function optimization, and analyze convergence properties of the algorithm. The crowding clustering genetic algorithm employs standard crowding strategy to form multiple niches and clustering operation to eliminate genetic drift. Numerical experiments on standard test functions indicate that crowding clustering genetic algorithm is superior to both standard crowding and deterministic crowding in quantity, quality and precision of multi-optimum search. The proposed algorithm is applied to the practical optimal design of varied-line-spacing holographic grating and achieves satisfactory results.

Keywords: Multimodal function optimization; Crowding clustering genetic algorithm; Evolutionary computation; Genetic drift; Varied-line-spacing holographic grating

1. Introduction

In practical optimization problems, objective functions often lead to multimodal domains. If the problem has several global optima or some local optima that might be good alternatives to the global optima, it is desirable to locate some of them during search process [1,2]. Traditional genetic algorithms (GAs) perform well in locating a single optimum but fail to provide multiple solutions. Various niching methods have been introduced into GAs to promote formation of stable sub-populations in neighborhood of optimal solutions. Therefore, multiple solutions can be identified at the end of optimization with certain extent of diversity.

There are many widely adopted niching techniques, such as standard crowding, deterministic crowding [3], sharing [4,5], clearing [6], dynamic niche clustering (DNC) [7,8], and so on. Standard crowding and deterministic crowding both suffer greatly from genetic drift, i.e. individuals are inclined to converge to several eminent solutions. In sharing and clearing, prior knowledge about fitness landscape is needed to set niche radius. Dynamic niche clustering uses clustering operation to set niche radius dynamically. But the set of cluster number will affect the quality and quantity of identified optimal solutions greatly.

In this paper, a novel multimodal optimization method combining crowding and clustering is proposed. Crowding clustering genetic algorithm (CCGA) employs a standard crowding strategy to form multiple niches and provide estimated landscape characteristics. On the other hand, clustering operation is intended to eliminate genetic drift and promote exploration in the whole fitness landscape. Both theoretical results and numerical experiments illuminate the satisfactory convergence properties of CCGA.

This paper is arranged as follows. Section 2 briefly reviews several main niching methods. CCGA is described in detail in Section 3 and its convergence properties are discussed in Section 4. In Section 5, the proposed algorithm is compared with standard crowding and deterministic crowding in several

* Corresponding author. Tel.: +86 551 3607454
E-mail addresses: qingling@mail.ustc.edu.cn (L. Qing), wug@ustc.edu.cn (W. Gang), ashesy@hkusua.hku.hk (Y. Zaiyue), quiping@ustc.edu.cn (W. Qiuping).

1568-4946/$ – see front matter
doi:10.1016/j.asoc.2006.10.014

multimodal functions. CCGA is applied to the practical optimization of varied-line-spacing holographic grating for the National Synchrotron Radiation Laboratory (NSRL) in Section 6. Section 7 provides conclusion and discussion of the future development of CCGA.

Without loss of generality, this paper deals with minimization problems only.

2. Niching techniques in multimodal optimization

2.1. Crowding

Standard crowding proposed by De Jong updates population through replacing similar parent [3]. For each child C, select CF (crowding factor) individuals in parent, choose the nearest parent P under some distance metric. If the objective function value of P is larger than that of C, use C to replace P, else preserve P. When CF is too small, standard crowding introduces great selection error. When CF equals to the number of population PopNum, selection error is eliminated but genetic drift still exists.

In deterministic crowding of Mahfoud [9], two parents P_1 and P_2 generate two children C_1 and C_2. For some distance metric D, if $D(P_1, C_1) + D(P_2, C_2)$ is smaller than $D(P_1, C_2) + D(P_2, C_1)$, then introduce competition between P_1 and C_1, P_2 and C_2, else compete between P_1 and C_2, P_2 and C_1. Deterministic crowding is a special standard crowding strategy where CF equals to 2, therefore selection error and genetic drift are both significant.

2.2. Sharing

Fitness sharing method modifies fitness landscape by reducing the payoff in densely populated regions [3]. For each individual, find all other individuals in its niche radius and share their fitness using the sharing function. The main disadvantage of fitness sharing is that it requires a niche radius related to the prior knowledge of objective function.

Clearing proposed by Petrowski is a special fitness sharing method. By comparing all individuals in the niche radius, several best individuals survive and the others are all cleared [6]. The disadvantage of clearing is similar to that of fitness sharing.

2.3. Other methods

Gan and Warwick combined clustering and fitness sharing and proposed a dynamic niche clustering algorithm [7,8]. For each generation, a clustering operation is executed with a dynamically updated cluster number. Among a cluster, fitness sharing is implemented with the niche radius provided by clustering. DNC algorithm does not require any prior knowledge of the objective function. But it is not an easy task to calculate the cluster number, which will affect the quantity and quality of identified optimal solutions greatly.

Other niching techniques, such as sequential niching, restricted tournament selection, dynamic niching, etc., all have their own merits and shortages [3]. But their application is relatively limited due to the existence of genetic drift, especially in the complicated multimodal function optimization problems without any prior knowledge of fitness landscape.

3. Crowding clustering genetic algorithm

3.1. Clustering in crowding algorithm

The basic idea in CCGA is using a standard crowding method to form niches and a clustering strategy to eliminate genetic drift. In fact, the idea of clustering has already been contained in crowding algorithm. For each child C_i, $i = 1, 2, \ldots,$ CF, assume that its nearest parent is P_j under a certain distance metric, we define C_i compete with P_j. Therefore, for each parent P_j, $j = 1, 2, \ldots,$ PopNum, there is a set of children $\{CS_j\}$ which compete with P_j. Then the $2 \times$ PopNum solutions which comprise parent and children form PopNum clusters $\{P_j, CS_j\}$, $j = 1, 2, \ldots,$ PopNum, and the fittest solution in each cluster survives to enter the next generation, as shown in Fig. 1. When CF < PopNum, selection error is inevitable. When CF = PopNum, selection error is eliminated. But in the same niche, several different clusters can coexist and evolve to the same optimal solution. This is the so-called genetic drift phenomena.

Crowding clustering genetic algorithm aims to prevent multiple clusters from converging to a single optimal solution. By introducing competition between clusters formed from a standard crowding model with CF = PopNum in the same niche, CCGA can avoid genetic drift effectively.

3.2. Procedure of CCGA

For a multimodal minimization problem:

$$\begin{aligned} \min \quad & f(x_1, x_2, \ldots, x_n) \\ \text{s.t.} \quad & x = [x_1, x_2, \ldots, x_n] \in Z \end{aligned} \tag{1}$$

Here, f is the objective function and Z is the feasible solution space. The procedure of CCGA is stated as follows:

Step 1. Initialize uniformly distributed population in feasible solution space. Number of population is PopNum.

Step 2. Recombine and mutate parents to generate PopNum children.

Step 3. For each parent P_j, $j = 1, 2, \ldots,$ PopNum, construct PopNum clusters $\{P_j, CS_j\}$ using a standard crowding model

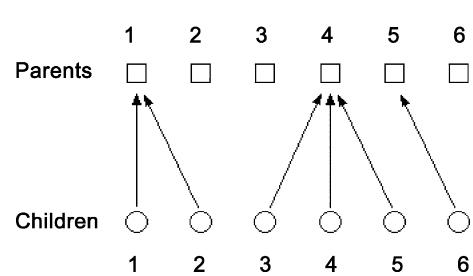

Fig. 1. Scheme of standard crowding. Six clusters are formed in this situation.

with CF = PopNum under a certain distance metric, as introduced in Section 3.1.

Step 4. For each cluster $\{P_j, CS_j\}$, select the fittest individual (individual with lowest objective value) CC_j as the *center of cluster*. The *objective value of cluster* CV_j is defined as the objective value of CC_j. For the other solutions in the cluster, calculate their distances from CC_j, select the largest distance as the *radius of cluster* CR_j.

Step 5. Sort all clusters according to their objective values in ascending order. Define a *set of reserved clusters* RC = Ø, each element in RC has a *center of cluster* RCC_i, and a *radius of cluster* RCR_i. For $j = 1, 2, \ldots$, PopNum, compare the jth cluster with all clusters RCC_i in RC, if for all i, $D(CC_j, RCC_i) > RCR_i$ or $\text{Peak}(CC_j, RCC_i) = 1$, then place CC_j into RC, and set the *radius of cluster* as $\min(CR_j, D(CC_j, RCC_i))$. Here, D() is the distance metric, Peak() is the *peak detection condition*. Here, we choose:

$$\text{Peak}(CC_i, RCC_j) = \begin{cases} 1, & f\left(\dfrac{CC_i + RCC_j}{2}\right) > \dfrac{f(CC_i) + f(RCC_j)}{2} \\ 0, & f\left(\dfrac{CC_i + RCC_j}{2}\right) \le \dfrac{f(CC_i) + f(RCC_j)}{2} \end{cases} \quad (2)$$

Step 6. Define the number of elements in *set of reserved clusters* RC as N_{RC}, generate (PopNum − N_{RC}) uniformly distributed individuals in the feasible solution space. These individuals and the centers of clusters in RC enter the next generation.

Step 7. Repeat step 2 to step 6 until the maximum generation number *MaxGen* reaches.

4. Theoretical analysis of CCGA

4.1. Convergence properties

In the minimization problem Eq. (1), let D: Z × Z → R be a certain distance metric in Z. For any M_i in Z, if $\forall x$ in Z, $\exists e > 0$: $D(x, M_i) < e \Rightarrow f(x) \ge f(M_i)$, M_i is defined as a *minimum* of $f(x)$. $M = \{M_1, \ldots, M_m\}$ is defined as the *minima set* of $f(x)$. For any M_j in M, if $\forall i$, $i = 1, 2, \ldots, m$, $f(M_i) \ge f(M_j)$, M_j is defined as a *global minimum* of $f(x)$ and the *global minima set* $G = \{G_1, \ldots, G_n\}$ of $f(x)$ contains all *global minimum* of $f(x)$. Any non-global minimum in M is defined as the *local minimum* L_i. $L = \{L_1, \ldots, L_{m-n}\}$ is defined as the *local minima set* of $f(x)$.

Now we introduce a new concept of *stable minimum* of $f(x)$. Firstly, all solutions in the *global minima set* G are elements in the *stable minima set* S. Secondly, for any L_k in L, if $\forall x$ in Z, $\exists S_i$ in S, $f(S_i) < f(L_k)$ such that $f(x) < f(L_k) \Rightarrow D(x, S_i) < D(x, L_k)$, L_k is called a *stable minimum* of $f(x)$ and is a element in S.

Property 1. *Assume that there are r stable minima in the minimization problem* Eq. (1), $n = \min(r, \text{PopNum})$. *If in some generation, the parents of* CCGA *occupy n best stable minima* $A = \{A_i, i = 1, 2, \ldots, n\}$, *then these parents will not be substituted by any child.*

Proof. For any A_i and any new generated child C_j, if $f(C_j) < f(A_i)$, then according to the definition of stable minima set, $\exists A_k$ in A, $f(A_k) < f(A_i)$ such that $D(C_j, A_k) < D(C_j, A_i)$. Therefore, each element in A will be the best one in corresponding cluster and will not be substituted by any child.

Property 2. *Assume that there are r stable minima in the minimization problem* Eq. (1), $n = \min(r, \text{PopNum})$. *If in some generation, the parents of* CCGA *occupy n best stable minima* $A = \{A_i\}$, $i = 1, 2, \ldots, n$, *and for any* A_i *in* A, $\forall x$ *in* Z, $f(A_i) \ge f(x) \Rightarrow \text{Peak}(x, A_i) = 1$, *then each element in* A *will not be cleared by any current parent.*

Proof. For any A_i in A and any P_j, $A_i \ne P_j$, if $f(A_i) < f(P_j)$, A_i will not be cleared by P_j. If $f(A_i) \ge f(P_j)$, from the assumption that $\forall x$ in Z, $f(A_i) \ge f(x) \Rightarrow \text{Peak}(x, A_i) = 1$, $\text{Peak}(A_i, P_j) = 1$, A_i will not be cleared by P_j. Therefore, each element in A will not be cleared by any current parent.

Property 3. *There is no more than one individual which converges to a single point in the minimization problem* Eq. (1).

Proof. Suppose there are two centers of clusters CC_i and CC_j, $CC_i = CC_j$, CC_i ranks before CC_j in the sorting operation. Then in the Step 5, $D(CC_i, CC_j) = 0$ and $\text{Peak}(CC_i, CC_j) = 0$. Therefore, CC_j is cleared by CC_i.

Remark 1. Property 1 indicates that stable minima solutions will not be affected by the children in the crowding algorithm. Generally speaking, a stable minimum has a relatively satisfying objective value, which is not excessively inferior to the objective value of any neighboring minimum.

Remark 2. Property 2 indicates that the fitness landscape which satisfies the peak detection condition can prevent stable minima from being affected by the parents in the clustering algorithm. The assumption of peak detection condition is a very critical requirement about fitness landscape. But in most cases, the radii of clusters are relatively small and the peak detection condition is not needed to protect other niches.

Remark 3. Summarizing Remarks 1 and 2, the n best stable minima cannot be affected by both children and parents under the assumptions above. Therefore, the set A is the absorption state of CCGA. In conclusion, CCGA uses the peak detection condition to protect the niches which contains the best stable minima.

Remark 4. Property 3 shows how CCGA eliminates genetic drift in the crowding algorithm. The competition between clusters prevents multiple individuals from residing in a single niche.

Remark 5. Property 3 also shows how CCGA preserves diversity of population. Each cleared individual will be regenerated and evolve forward, until it reaches an unexplored niche.

4.2. Peak detection condition

The concept of *peak detection condition* is similar to the *hill–valley function* proposed by Ursem [11]. In minimization problem in Eq. (1), a hill–valley function can be calculated for a series of points $\{z_1, z_2, \ldots, z_n\}$ in the line segment between x_1 and x_2:

$$\mathrm{HV}(x_1, x_2) = \prod_{i=1}^{n} \mathrm{HV}_i \tag{3}$$

$$\mathrm{HV}_i = \begin{cases} 1, & f(z_i) > \max(f(x_1), f(x_2)) \\ 0, & f(z_i) \le \max(f(x_1), f(x_2)) \end{cases} \tag{4}$$

When the number of sample points $\{z_1, z_2, \ldots, z_n\}$ increases, computation time in objective function evaluation will increase proportionally. Therefore, we use only one sample point $(x_1 + x_2)/2$ here:

$$\mathrm{HV}(x_1, x_2) = \begin{cases} 1, & f\left(\dfrac{x_1 + x_2}{2}\right) > \max(f(x_1), f(x_2)) \\ 0, & f\left(\dfrac{x_1 + x_2}{2}\right) \le \max(f(x_1), f(x_2)) \end{cases} \tag{5}$$

Hill–valley function never returns 1 when two individuals are in the same niche. Peak detection condition is possible to return 1 at that time, but it runs less risk of returning 0 when two individuals are in different niches, because:

$$f\left(\frac{x_1 + x_2}{2}\right) > \max(f(x_1), f(x_2)) \Rightarrow f\left(\frac{x_1 + x_2}{2}\right) > \frac{f(x_1) + f(x_2)}{2} \tag{6}$$

Eq. (6) means that if HV() equals to 1 then Peak() equals to 1. According to Remark 2, we tend to protect current clusters with Peak() = 1, therefore peak detection condition is more proper here than hill–valley function here, as seen in Fig. 2.

Actually, there are many other forms of peak detection conditions by varying sample point or varying scaling method. The choice of the different peak detection condition reflects the different estimation of fitness landscape and influences the optimization results greatly.

4.3. Settings in CCGA

From description in Section 3.2, we can see that CCGA is a simple combination of standard crowding and clustering method and no prior knowledge of fitness landscape is needed. There are only two control parameters PopNum and MaxGen and their roles are easy to understand.

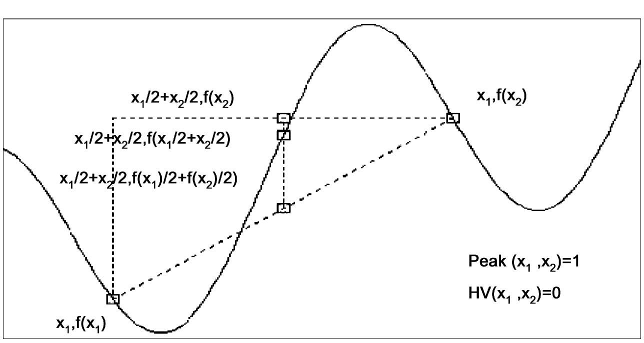

Fig. 2. Demonstration of peak detection condition and hill–valley function.

Here, we are going to discuss several detailed strategies in CCGA, such as crossover, mutation, boundary handling, and distance metric.

There are various crossover and mutation strategies in genetic algorithms. We adopt the linear crossover with random parameter for real coded GA [10]. We do not use any mutation operation here for convenience because crossover operation has introduced randomness already.

In dealing with solutions which exceed the boundary of feasible solution space, we simply place them in the nearest boundary other than randomly regeneration.

CCGA is originally proposed for real coding. Therefore Euclidean distance metric is a natural choice. When CCGA is adapted for binary coding, there are much more choices, such as genotype metrics, phenotype metrics, and so on.

5. Numerical results

5.1. Performance criteria

We evaluate the multimodal optimization algorithm in three aspects: quantity, quality, and precision of solutions.

Quantity: Number of detected ‘‘valleys’’ in feasible solution space. In other words, run a certain local search using each solution as the initial point, calculate the number of different minima in feasible solution space, and name it as *niche number* NN. Here, we use simplex method as a local search operator.

Quality: Sum of objective values of the corresponding minima in all detected niches, named as *niche quality* NQ. NQ/NN indicates the average quality of niches.

Precision: Sum of objective function values of the best solutions in all detected niches, named as *niche precision* NP. NP/NN indicates the average precision of niches.

5.2. Test environment

There are three multimodal function optimization algorithms to compare: (a) CC, crowding clustering GA; (b) SC, standard crowding GA with CF = PopNum; (c) DC, deterministic crowding GA. Public parameters of algorithms are: population number PopNum = 40, maximum generation number MaxGen = 100. Strategies of crossover, mutation, boundary handling and distance metric are as introduced in Section 4.3.

Two standard test functions are used here. The first one is Ackley function [2], which has more than 40 global and local minima:

$$\begin{aligned} \min \quad & f_1(x, y) = 20 - 20 \exp\left(-0.2\sqrt{\frac{(x^2 + y^2)}{2}}\right) - \exp\left(\frac{\cos(2\pi x) + \cos(2\pi y)}{2}\right) + e \\ \text{s.t.} \quad & -30 \le x, y \le 30 \end{aligned} \tag{7}$$

The second one is Six-Hump Camel function [2], which has two global minima, two stable local minima, two unstable local minima:

$$\begin{aligned} \min \quad & f_2(x,y) = \left(4 - 2.1\,x^2 + \frac{x^4}{3}\right)x + xy + (-4 + 4\,y^2)y^2 \\ \text{s.t.} \quad & -1.9 \le x \le 1.9,\ -1.1 \le y \le 1.1 \end{aligned} \tag{8}$$

5.3. Test results

The typical optimization results of the three algorithms on test functions are shown in Figs. 3 and 4. Run the three algorithms on test functions 10 times, and calculate the mean values and variances of each criterion, as shown in Tables 1 and 2.

In both two functions, DC can find only one minimum due to selection error and genetic drift. SC finds fewer minima than CC, which eliminates the genetic drift in SC with clustering operator. The criteria NQ/NN and NP/NN of CC are worse than those of SC, it is because that CC finds more niches than SC.

It is interesting to consider the Six-Hump Camel function. DC converges to a single global minimum quickly. SC converges to equal or less than four stable minima, and then never reaches the unstable minima. But CC can still reach the unstable minima after converges to four stable minima. This phenomenon shows that CC can prevent genetic drift, and preserve diversity efficiently.

6. CCGA in holographic grating design

6.1. Varied-line-spacing holographic grating design

Holographic gratings are the main diffractive elements in vacuum ultraviolet and soft X-ray spectrum. In recent years, varied-line-spacing (VLS) holographic gratings have been widely used in high-resolution spectrometers and monochromators due to their excellent self-focusing and aberration-eliminating properties [12,13]. Application of VLS holographic gratings will reduce the number of auxiliary optical elements, simplify the structure of diffractive instruments, and increase the light throughput and optical resolution efficiently.

The schematic recording optical system of VLS holographic grating is shown in Fig. 5 [14]. It consists of two coherent light sources C and D, two spherical mirrors M_1 and M_2, and a spherical grating blank G. Given the radii of M_1, M_2 and G, and the recording wavelength λ_0, the groove shape of G is decided by eight recording parameters: η_C, γ, η_D, δ, p_C, p_D, q_C, q_D.

The aim of optimal design is to find several sets of recording parameters to form the expected groove shape, which is generally represented by the groove density in the *Y*-axis:

$$\begin{aligned} n_e &= n_0(1 + b_2 w + b_3 w^2 + b_4 w^3) \\ &= n_0 + n_0 b_2 w + n_0 b_3 w^2 + n_0 b_4 w^3 \end{aligned} \tag{9}$$

While the practical groove density can be calculated from a complicated function *f* of recording parameters:

$$n_p = f(\eta_C, \gamma, \eta_D, \delta, p_C, p_D, q_C, q_D) \tag{10}$$

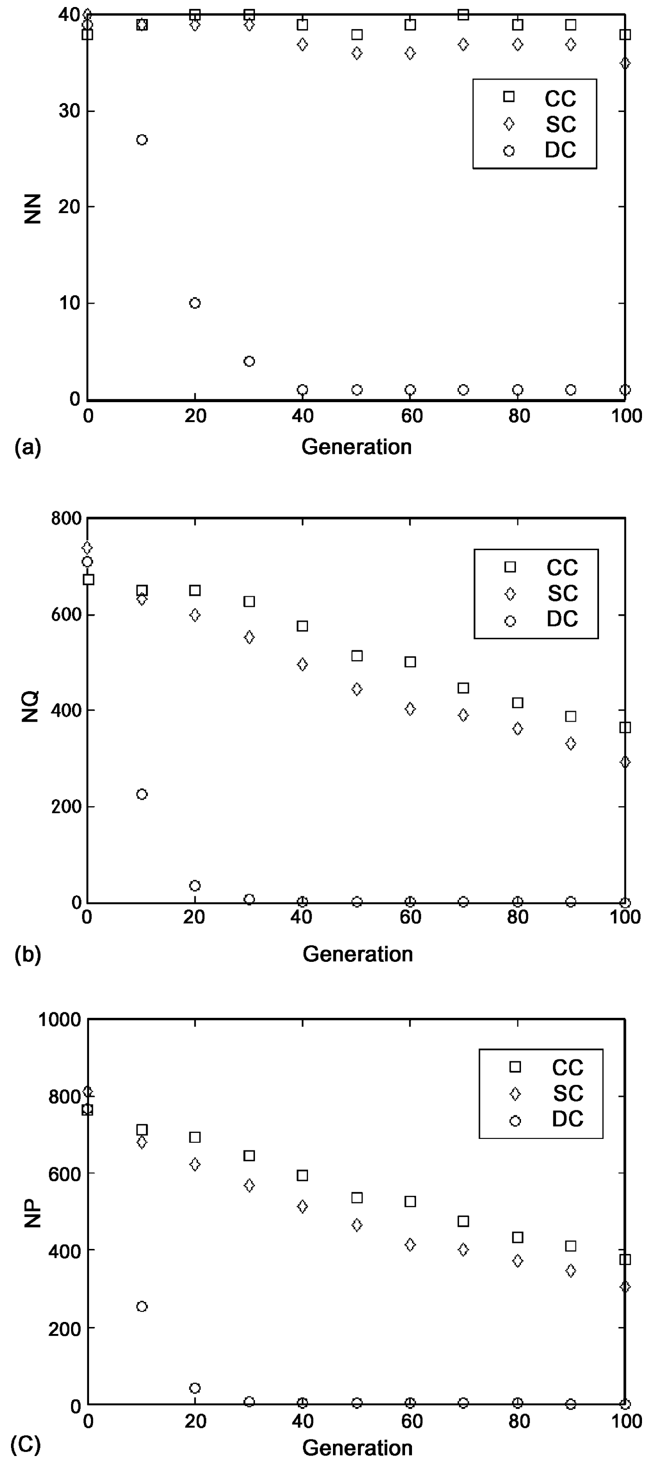

Fig. 3. (a) Comparison of niche number of three algorithms on Ackley. (b) Comparison of niche quality of three algorithms on Ackley. (c) Comparison of niche precision of three algorithms on Ackley.

Consider minimizing the integration of square error of groove density in *Y*-axis [15]:

$$\min \quad J = \int_{-w_0}^{w_0} (n_p - n_e)^2\, \mathrm{d}w \tag{11}$$

where w_0 is the half-width of the grating to record.

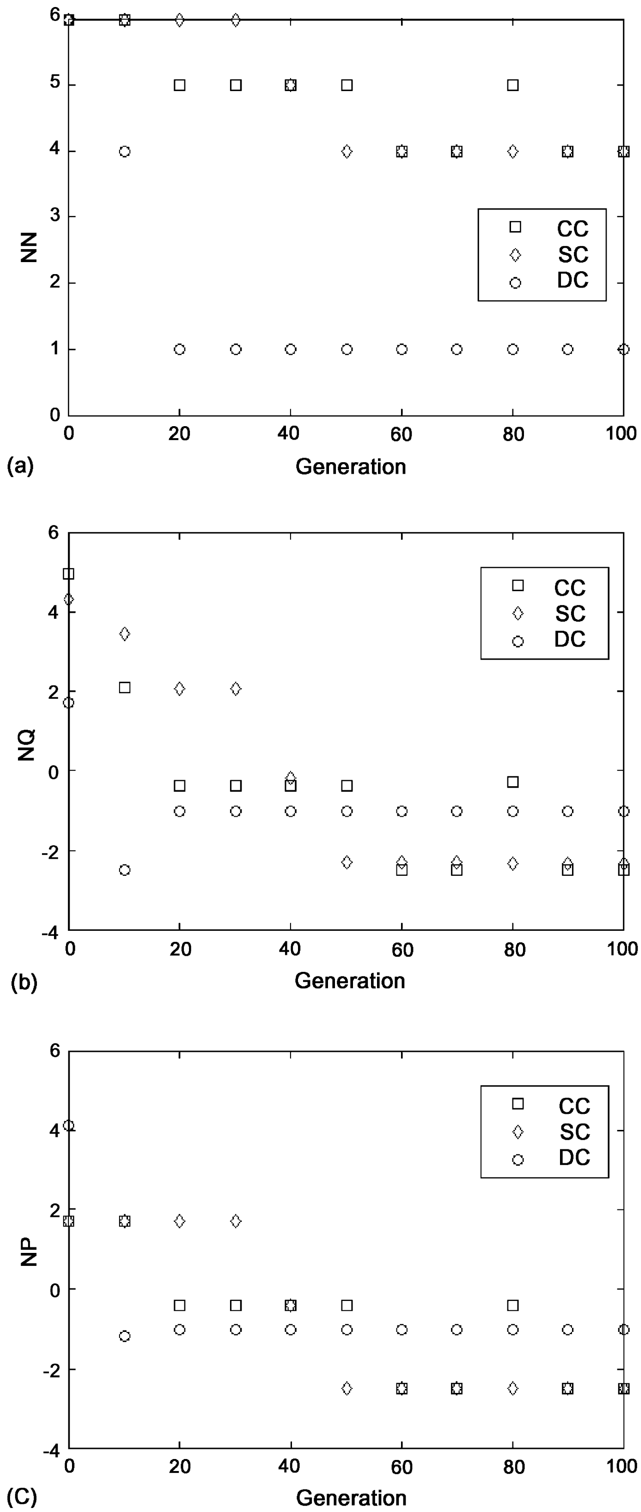

Fig. 4. (a) Comparison of niche number of three algorithms on Six-Hump Camel. (b) Comparison of niche quality of three algorithms on Six-Hump Camel. (c) Comparison of niche precision of three algorithms on Six-Hump Camel.

It is a basic requirement of recording parameter optimization to locate multiple solutions. Because in practical recording system, there are many other auxiliary optical elements, such as beam splitter, light filter, and so on, which are not shown in the schematic diagram. If the recording parameters are not properly selected, these auxiliary elements will interfere with each other. But this kind of constraints is difficult to express in a mathematical form. Therefore, it is reasonable to provide the optical engineers with multiple solutions for trials.

Table 1
Average test results of three algorithms on Ackley

	NN		NQ/NN		NP/NN	
	Mean	Variance	Mean	Variance	Mean	Variance
CC	38.4000	2.4000	8.8173	0.8055	9.2027	0.8005
SC	35.2000	5.6000	7.9321	0.3309	8.2280	0.3251
DC	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 2
Average test results of three algorithms on Six-Hump Camel

	NN		NQ/NN		NP/NN	
	Mean	Variance	Mean	Variance	Mean	Variance
CC	4.4000	0.2667	−0.4053	0.0794	−0.3992	0.0817
SC	3.9000	0.1000	−0.6371	0.0000	−0.6345	0.0002
DC	1.0000	0.0000	−1.0316	0.0000	−1.0316	0.0000

6.2. Holographic grating design with CCGA

For the VLS planar holographic grating in soft X-ray magnetic circular dichroism (SXMCD) beamline in National Synchrotron Radiation Laboratory (NSRL), the groove density is described as:

$$n_0 = 1.4000 \times 10^3 (\mathrm{line/mm}) \tag{12.1}$$

$$b_2 = 8.2453 \times 10^{-4} (1/\mathrm{mm}) \tag{12.2}$$

$$b_3 = 3.0015 \times 10^{-7} (1/\mathrm{mm}^2) \tag{12.3}$$

$$b_4 = 0.0000 \times 10^{-10} (1/\mathrm{mm}^3) \tag{12.4}$$

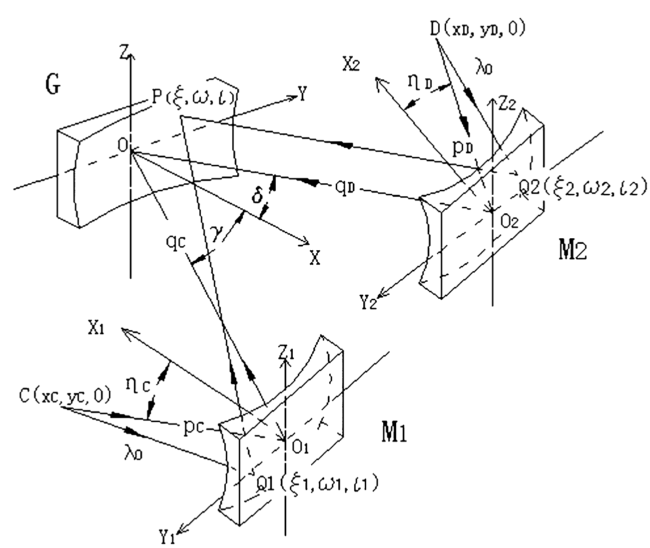

Fig. 5. Schematic diagram of recording holographic grating.

Table 3
Recording parameters from CCGA and consequential simplex search

	γ (rad)	η_C (rad)	δ (rad)	η_D (rad)	p_C (m)	q_C (m)	p_D (m)	q_D (m)
1	0.3082	−1.3339	1.0794	0.0269	0.3124	1.6512	1.4833	2.3038
2	−0.0765	0.2291	0.5258	1.1235	0.7871	1.9206	0.4583	1.1105
3	0.4292	−0.9632	1.4662	0.0188	0.0136	1.7104	0.4376	0.3658

Half-width w_0 equals to 90 mm. The radii of auxiliary spherical mirrors are both 1000 mm and the recording wavelength is 413.1 nm.

In this paper, CCGA is used to optimize the recording parameters, with PopNum = 100 and MaxGen = 200. At the end of CCGA, a consequential simplex local search method is applied to each solution to improve the optimization precision. Table 3 lists the best three sets of recording parameters and the corresponding groove density is shown in Table 4.

Table 4
Corresponding groove density of optimized recording parameters

	n_0 (line/mm)	b_2 (1/mm)	b_3 (1/mm^2)	b_4 (1/mm^2)
1	1.4000×10^3	8.2456×10^{-4}	3.0014×10^{-7}	8.2387×10^{-14}
2	1.4000×10^3	8.2506×10^{-4}	3.0296×10^{-7}	1.3889×10^{-11}
3	1.4001×10^3	8.2463×10^{-4}	2.9836×10^{-7}	-1.9006×10^{-11}

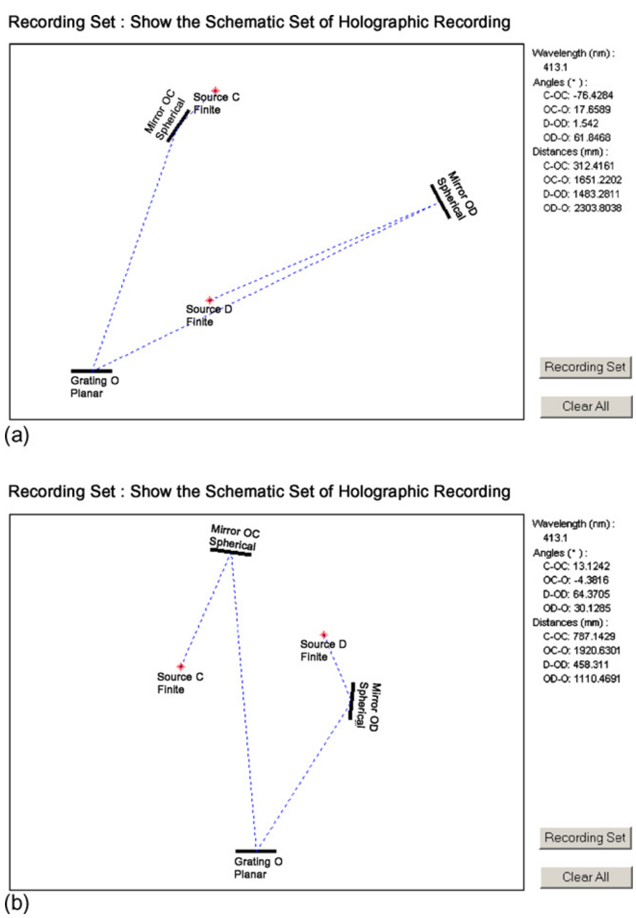

Fig. 6. (a) Schematic diagram with recording parameters in group 1. (b) Schematic diagram with recording parameters in group 2.

Group 1 is a typical set of recording parameters which is superior in objective function value but does not satisfy the implicit constraints. As shown in Fig. 6(a), incident arm p_D and reflective arm q_D are too close to realize in practical adjustment. In another example of group 3, incident arm p_C is very small. Therefore, it is not a reasonable solution in the fabrication too.

On the contrary, recording parameters of group 2 perform well in both achieving the groove density requirement and satisfying the implicit constraints. The optical schematic diagram is shown in Fig. 6(b).

7. Conclusion

In this paper, we propose a novel crowding clustering genetic algorithm (CCGA) for multimodal function optimization. By combining standard crowding and clustering, CCGA conquers both genetic drift in standard crowding and niche radius problem in clustering algorithm. The *peak detection condition* links crowding and clustering, and prevents niches from being destroyed by mistake. On the other hand, CCGA does not need any prior knowledge of fitness landscape in the optimization process. Therefore, CCGA is a proper choice in optimization of complicated multimodal functions.

The convergence properties of CCGA are analyzed theoretically in this paper. Numerical results on standard test functions indicate that CCGA is superior to both standard crowding and deterministic crowding algorithm in quantity, quality and precision of multi-optimum search. The successful application in the practical varied-line-spacing holographic grating design problem illustrates the satisfactory multimodal search ability of CCGA.

Future development of CCGA will focus on improvement of its convergence rate. For example, to combine crowding clustering with evolutionary algorithms of rapid convergence rate, such as differential evolution [2]. Furthermore, competition between clusters can be introduced as in the restricted evolution model [15,16] to improve both convergence rate and niche quality of CCGA.

References

[1] D. Beasley, D. Bull, Martin R, A sequential niche technique for multimodal function optimization, Evolut. Comput. 1 (2) (1993) 101–125.
[2] R Thomsen, Multimodal optimization using crowding based differential evolution, in: Proceedings of IEEE Congress on Evolutionary Computation, 2004, pp. 1382–1389.
[3] B. Sareni, Krahenbuhl L, Fitness sharing and niching methods revisited, IEEE Trans. Evolut. Comput. 2 (3) (1998) 97–106.

[4] D. Goldberg, J. Richardson, Genetic algorithms with sharing for multimodal function optimization, in: Proc. Int. Conf. Genetic Algorithms, 1987, 41–49.
[5] D. Goldberg, K. Deb, Horn J, Massive multimodality, deception and genetic algorithms, Parallel Problem Solving from Nat. 2 (1992) 37–46.
[6] A. Petrowski, A clearing procedure as a niching method for genetic algorithms, in: Proceedings of IEEE International Conference on Evolutionary Computation, 1996, pp. 798–803.
[7] J. Gan, K. Warwick, A genetic algorithm with dynamic niche clustering for multimodal function optimization, in: Proceedings of International Conference on Artificial Neural Network and Genetic Algorithms, 1999, pp. 248–255.
[8] J. Gan, K. Warwick, Dynamic niche clustering: a fuzzy variable radius niching technique for multimodal optimisation in GAs, in: Proceedings of IEEE Congress on Evolutionary Computation, 2001, pp. 215–222.
[9] S. Mahfoud, Crossover interactions among niches, in: Proceedings of IEEE International Conference on Evolutionary Computation, 1994, pp. 188–193.
[10] A. Chipperfield, P. Fleming, The MATLAB genetic algorithm toolbox, in: Proceedings of IEE Colloquium on Applied Control Techniques Using MATLAB, Digest No. 1995/014, 1995.
[11] R. Ursem, Multinational evolutionary algorithms, in: Proceedings of IEEE Congress on Evolutionary Computation, 1999, pp. 1633–1640.
[12] S Singh, Diffraction gratings: aberrations and applications, Opt. Laser Technol. 30 (1999) 195–218.
[13] M. Koike, T. Yamazaki, Y Harada, Design of holographic gratings recorded with aspheric wavefront recording optics for soft X-ray flat-field spectrographs, J. Electron Spectrosc. Relat. Phenom. 101–103 (1999) 913–918.
[14] T. Namioka, Koike M, Aspherical wave-front recording optics for holographic gratings, Appl. Opt. 34 (13) (1995) 2180–2186.
[15] Q. Ling, G. Wu, Q. Wang, Restricted evolution based multimodal function optimization in holographic grating design, in: Proceedings of IEEE Congress on Evolutionary Computation, 2005, pp. 789–794.
[16] C. Im, H. Kim, H. Jung, et al., A novel algorithm for multimodal function optimization based on evolution strategy, IEEE Trans. Magnet. 40 (2) (2004) 1224–1227.