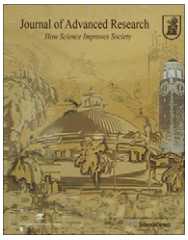
Journal of Advanced Research (2010) 1, 301–307

Cairo University

Journal of Advanced Research



ORIGINAL ARTICLE

Context based clearing procedure: A niching method for genetic algorithms

Magda B. Fayek, Nevin M. Darwish, Mayada M. Ali *

Computer Engineering Department, Faculty of Engineering, Cairo University, Giza, Egypt

Received 27 July 2009; revised 31 January 2010; accepted 7 March 2010
Available online 13 October 2010

KEYWORDS

Genetic algorithms;
Multimodal optimization;
Niching methods;
Niching problem

Abstract In this paper we present CBC (context based clearing), a procedure for solving the niching problem. CBC is a clearing technique governed by the amount of heterogeneity in a subpopulation as measured by the standard deviation. CBC was tested using the M7 function, a massively multimodal deceptive optimization function typically used for testing the efficiency of finding global optima in a search space. The results are compared with a standard clearing procedure. Results show that CBC reaches global optima several generations earlier than in the standard clearing procedure. In this work the target was to test the effectiveness of context information in controlling clearing. A subpopulation includes a fixed number of candidates rather than a fixed radius. Each subpopulation is then cleared either totally or partially according to the heterogeneity of its candidates. This automatically regulates the radius size of the area cleared around the pivot of the subpopulation.

Introduction

A simple genetic algorithm [1] (SGA) is a known algorithm for searching the optimum of unimodal functions in a bounded search space. However, SGA cannot find the multiple global maxima of a multimodal function [1,2]. This limitation in the performance of the SGA has been overcome by a mechanism that creates and maintains several subpopulations within the search space. The goal of each of these subpopulations is to lead to one of the optimum maxima. In such a way that each of the highest maxima of the multimodal function can attract one of the optima. These mechanisms are referred to as ''niching methods'' [2].

We list below some of the famous niching methods: Simple iteration runs the simple GA several times to the same problem and the results of the particular runs are collected. Fitness Sharing [3] reduces the fitness of an individual if there are many other individuals similar to it and so the GA is forced to maintain diversity in the population. In the Sequential Niche Technique [4] the GA is run many times on the same problem. After every run the optimized function is modified (multiplied by a derating function) so that the optimum just

* Corresponding author. Tel.: +20 2 38955450/+20 112579060.
E-mail address: mayadahadhoud@gmail.com (M.M. Ali).

Peer review under responsibility of Cairo University.
doi:10.1016/j.jare.2010.09.001

Production and hosting by Elsevier

found will not be located again. In Pétrowski [5] a clearing procedure is introduced. In this approach subpopulations are determined according to certain similarity measures. Those subpopulations are then cleared to allow evolution of other optima. We will refer to this procedure as the standard clearing procedure. In Cioppa et al. [6] a Dynamic Fitness Sharing algorithm is introduced to overcome the limitations of the ordinary Fitness Sharing algorithm, which are the lack of an explicit mechanism for identifying or providing any information about the location of the peaks in the fitness landscape, and the definition of species implicitly assumed by Fitness Sharing. In Ellabaan and Ong [7] a Valley-Adaptive Clearing Schema is introduced, which comprises three core phases: *the valley identification phase* categorizes the population of individuals into groups of individuals sharing the same valley; the dominant individual (i.e., in terms of fitness value) of a valley group is archived if it represents a unique local optimum solution, while all other members of the same group undergo *the valley replacement phase* where relocation of these individuals to new valleys are made so that unique local optimum solution elsewhere may be uncovered; in the event that no local optimum solution exists in a valley group, all individuals of the group will undergo *the valley clearing stage* where elite individuals are ensured to survive across the search generation while all others are relocated to new basin of the attractions. In Shir and Bäck [8,9] a Dynamic Niching Algorithm is introduced.

The algorithms described in the previous section work on the assumption that maxima are evenly distributed throughout the search space, but actually they are not. Some approaches take this distribution into consideration by using a variable radius to fit the subpopulations to be cleared. An example is the GAS (*GA Species*) algorithm [10], where a radius function is used instead of a fixed radius, and the UEGO (*Universal Evolutionary Global Optimization*) algorithm [11,12]. These approaches require additional processing for estimating the number of candidates to be cleared.

In this paper we introduce a new niching technique: the Context Based Clearing (CBC) procedure. CBC uses a fixed number of candidates in a clearing subpopulation rather than a fixed radius. Unlike the standard clearing procedure the CBC procedure makes use of local information to guide the clearing procedure. In addition, it avoids additional processing overhead by using a fixed radius. Tests have shown that CBC rapidly finds a subset of solutions for the tested multimodal function.

In the next part of this paper the standard clearing procedure is explained. Then the proposed CBC procedure is presented. In part 3, the proposed CBC technique is compared with the standard clearing procedure in terms of their relative complexity.

Finally, the results of applying the CBC procedure on a multimodal deceptive function are given and discussed with respect to the standard clearing procedure.

Methodology

Description of standard clearing procedure

The clearing procedure is a niching method inspired by the niching principle [13], namely, the sharing of limited resources within subpopulations of individuals characterized by some similarities. However, instead of evenly sharing the available resources among the individuals of a subpopulation, the clearing procedure supplies these resources only to the best individuals of each subpopulation.

In the clearing procedure each subpopulation contains a dominant individual (winner), which is the one with the best fitness in the subpopulation. An individual belongs to a given subpopulation if its dissimilarity with the winner of the subpopulation is less than a given threshold, the clearing radius. The fitness of the dominant individual is preserved while the fitness of all the other individuals of the same subpopulation is set to zero.

Hence, for a given population, a unique set of winners will be produced. The same mechanism is applied for each population. Thus a list of all winners is produced over a run.

The proposed CBC procedure

The CBC procedure is a clearing procedure that makes use of context information to prevent clearing candidates that may lead to significant optima. Context refers in our case to the fitness distribution within a certain area around pivot elements, as explained below. Within the same area, if candidates have similar fitness, it is safe to clear the complete area as then all candidates belong to the same optima. However, if candidates' fitness differs significantly (which is measured by the standard deviation, as will be shown), it may cause loss of important data if the whole set of candidates is cleared.

CBC is embedded within GA, as shown in Fig. 1. It begins after evaluating the fitness of the individuals and before applying selection and crossover.

The CBC procedure performs clearing according to the heterogeneity of the individuals within the subpopulation, where heterogeneity is measured using the standard deviation of individuals' fitness.

Each subpopulation has a pivotal individual, which is the individual with the highest fitness. The number of individuals in a subpopulation around a certain pivot is determined by the amount of similarity between individuals and the pivot. Similarity can be estimated using the Hamming distance for

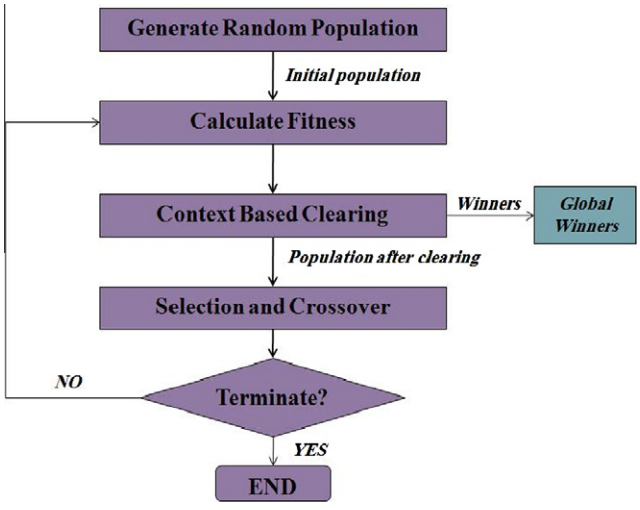


Fig. 1 Basic steps of the CBC procedure.

binary coded genotypes, the Euclidian distance for real coded genotypes or any other defined measure.

CBC procedure

The CBC procedure uses a number of parameters, as follows:

– *Subpopulation_Percentage* (*SP*): determines the proportion of individuals that fall within a subpopulation. These individuals are those nearest to the pivot of the subpopulation with respect to distance. The number of individuals within each subpopulation is calculated as follows:

$$M = SP * population\_size/100 \quad (1)$$

– *Niche Radius*: the threshold value for clearing candidates around the pivot in case of insufficient homogeneity of subpopulation.

For each subpopulation the standard deviation of all individuals' fitness within the subpopulation is calculated. Dependent on this value some actions will be taken.

Fig. 2 shows the general steps of the CBC procedure.

The CBC procedure runs as follows:

First, individuals of the whole population are sorted in descending order with respect to their fitness into a *candidate queue*.

Secondly, a subpopulation is created as follows:

The highest fitness candidate in the candidate queue is selected as a pivot.

1. Select pivot neighbors within the subpopulation around the pivot as determined by the *SP* parameter.
2. Calculate the standard deviation of fitness values for the subpopulation to specify the heterogeneity among candidates of the subpopulation.
3. If the standard deviation value is less than the given threshold (which ranges between minimum fitness value and maximum fitness value of the subpopulation candidates, and is empirically estimated), then candidates in this subpopulation will be cleared by setting their fitness to zero in the subpopulation as well as in the candidate queue. Otherwise, only those candidates within a distance less than or equal to the Niche Radius with respect to the pivot will be cleared (again in the subpopulation as well as in the candidate queue).
4. The next candidate with fitness > zero in the candidate queue is taken as pivot. Then steps 1–4 are repeated.

The winners of all subpopulations are stored in the global winners array. The result of the CBC procedure is a set of the cleared individuals and the winners with fitness > average fitness (of winners' fitness value). This population enters crossover and mutation stage to generate the next new population.

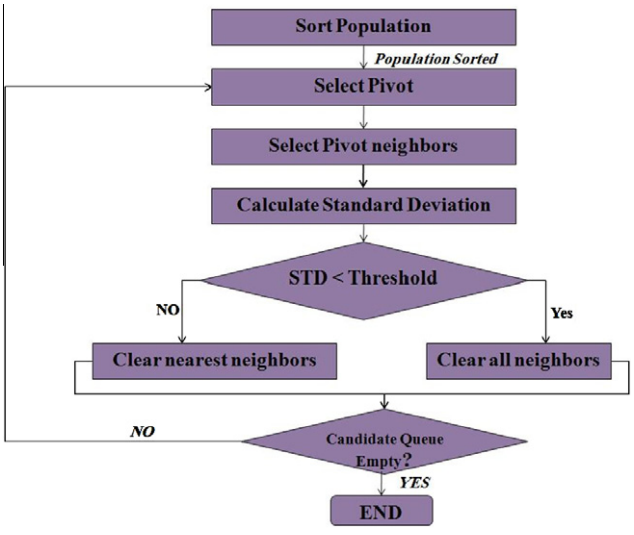


Fig. 2 CBC procedure flow chart.

Results and discussion

To test CBC the M7 function [2,14], typically applied in testing the capability of search techniques to locate global maxima, has been used.

The M7 function is defined as follows:

$$\mathrm{M7}(x_0, \ldots, x_{29}) = \sum_{i=0}^{4} u\left(\sum_{j=0}^{5} x_{6i+j}\right) \quad (2)$$

where $\forall k,\ x_k \in \{0, 1\}$. Function $u(x)$ is defined for the integer values 0–6 (Fig. 3). It has two maxima of value 1 at the points $x = 0$ and $x = 6$, as well as a local maximum of value 0.640576 for $x = 3$. Function u has been specifically built to be deceptive.

Function M7 has 32 global maxima of value equal to 5 (e.g. 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1, 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0), and several million local maxima, the values of which are between 3.203 and 4.641.

Experimental settings

The parameters used in the GA [5,15] are: *Population Size* equal to 600 binary coded genotypes of 30 bits, single point cross over with crossover rate equal to 1, standard binary mutation with mutation rate equal to 0.002, tournament selection method, Hamming distance used as a dissimilarity measure between genotypes normalized so that the biggest value in the search domain of the GA is equal to 1, the *Threshold* value is taken as equal to 0.25, and the *Clearing Radius* is taken

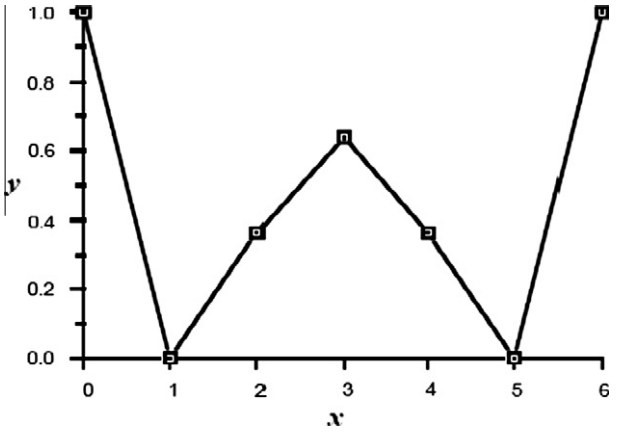


Fig. 3 The M7 function ($u(x)$).

as equal to 0.2, which corresponds to the smallest distance that exists between two global maxima [5].

To study the effect of the *SP* parameter on the number of detected optima, the *SP* parameter was set to the values 5, 10, 20 and 50.

Detailed results are presented in the following section.

CBC results

The performance of CBC is measured by the mean number of peaks found by the GA at a given generation g in 10 different runs using the same parameters. A peak is found if at least one individual in the population corresponds to a global maximum.

Figs. 4–7 shows the effect of changing the value of the *SP* parameter (5, 10, 20 and 50) on performance. For all *SP* values, the first global maximum was found at generation 5.

Compared to the standard clearing procedure as reported in Pétrowski [5] and given in Figs. 9 and 10, CBC reaches a solution several generations earlier. In addition, it is noted that changing the value of the *SP* parameter affects the maximum number of peaks found, and also affects processing time.

Increasing the value of the *SP* parameter decreases the maximum number of solutions found. This is due to the fact that a large subpopulation size is more probable to suppress some optima. Obviously, increasing the value of the *M* parameter increases the processing time. As *M* grows larger it would finally reach *N*, which is the case in the standard clearing procedure.

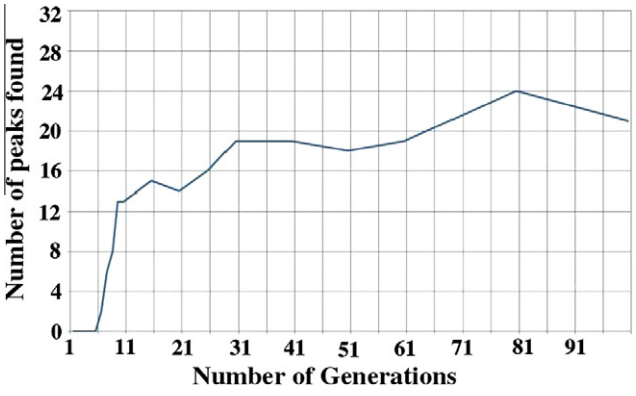


Fig. 4 CBC results (*SP* = 5).

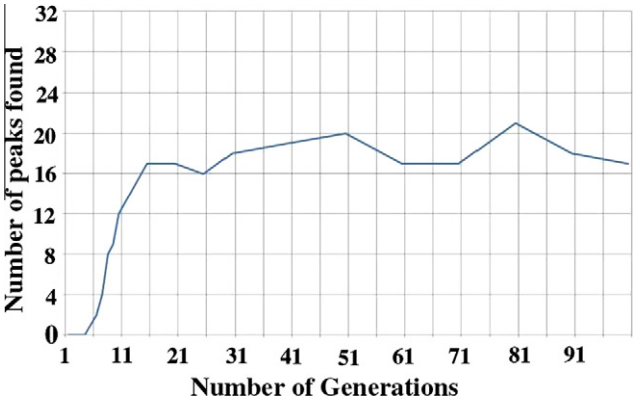


Fig. 5 CBC results (*SP* = 10).

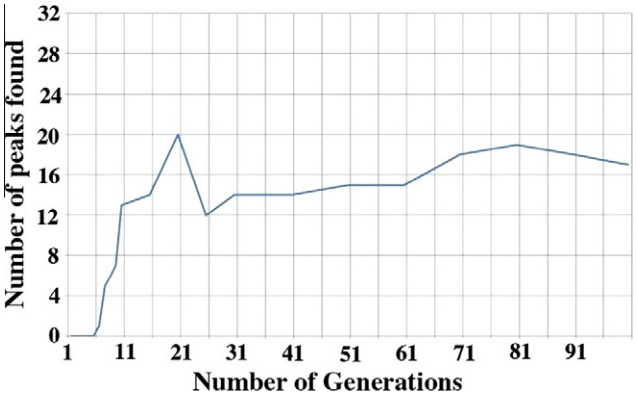


Fig. 6 CBC results (*SP* = 20).

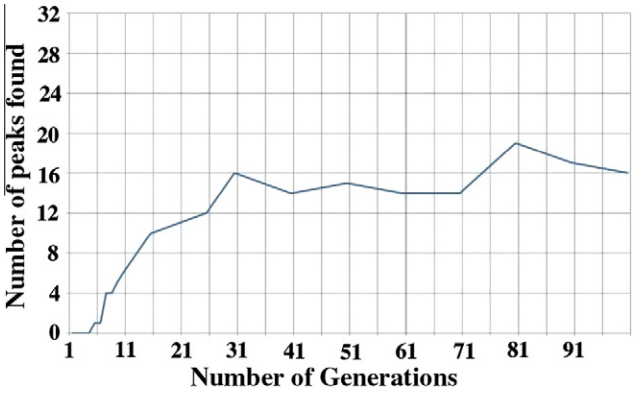


Fig. 7 CBC results (*SP* = 50).

Table 1 Number of peaks for different subpopulation sizes.

SP	5	10	20	50
# of peaks	24	21	20	18

Table 1 shows the average total number of distinct peaks found over a run using different values of the *SP* parameter.

According to the previous table, increasing the value of *SP* decreases the maximum number of peaks found; the best results found are for subpopulation size *Sp* = 5% and 10% where the processing time is nearly the same; but when the subpopulation size was increased to 50% the number of detected peaks decreased and the processing time increased.

Fig. 8 shows the effect of changing the value of the population size parameter (100, 300, 600 and 800) on performance.

As noted, changing the value of the population size parameter affects the number of optima reached; for different population sizes CBC procedure reaches the first optimum nearly at the same generation, which makes it a stable algorithm.

CBC vs. standard clearing results

In this section the performances of the CBC procedure and the standard clearing procedure are compared. The comparison

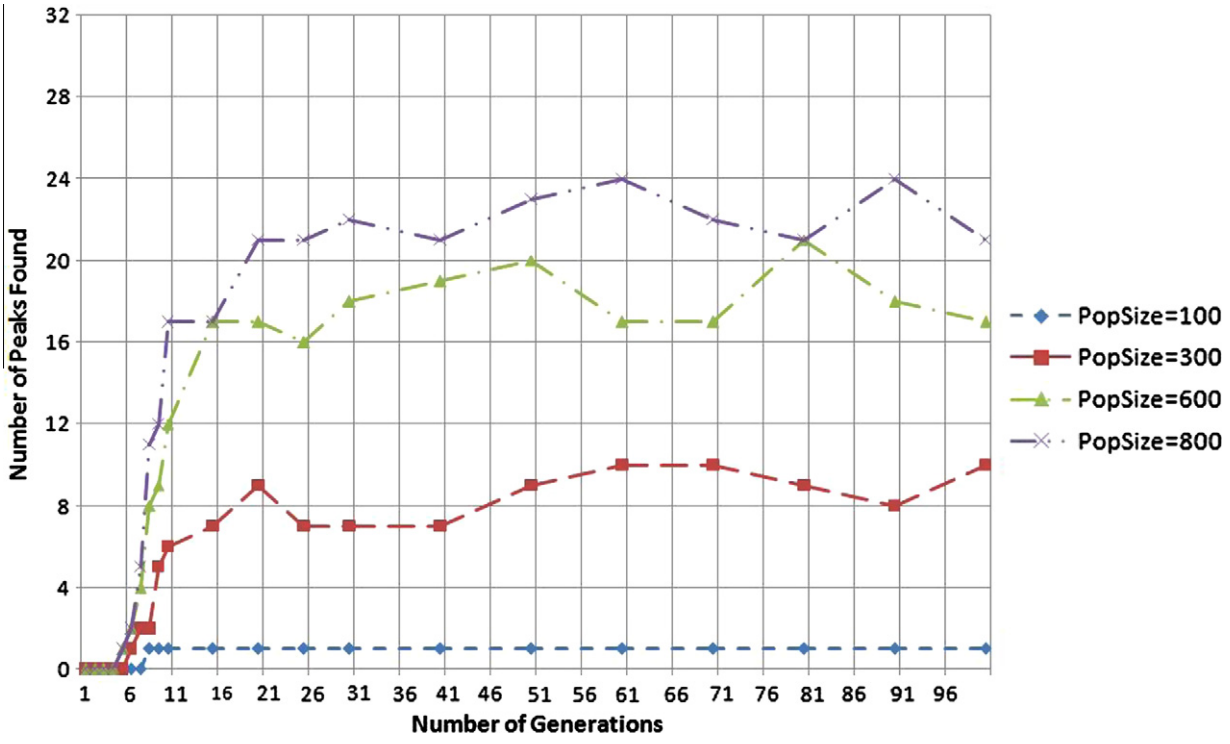


Fig. 8 CBC with different population sizes.

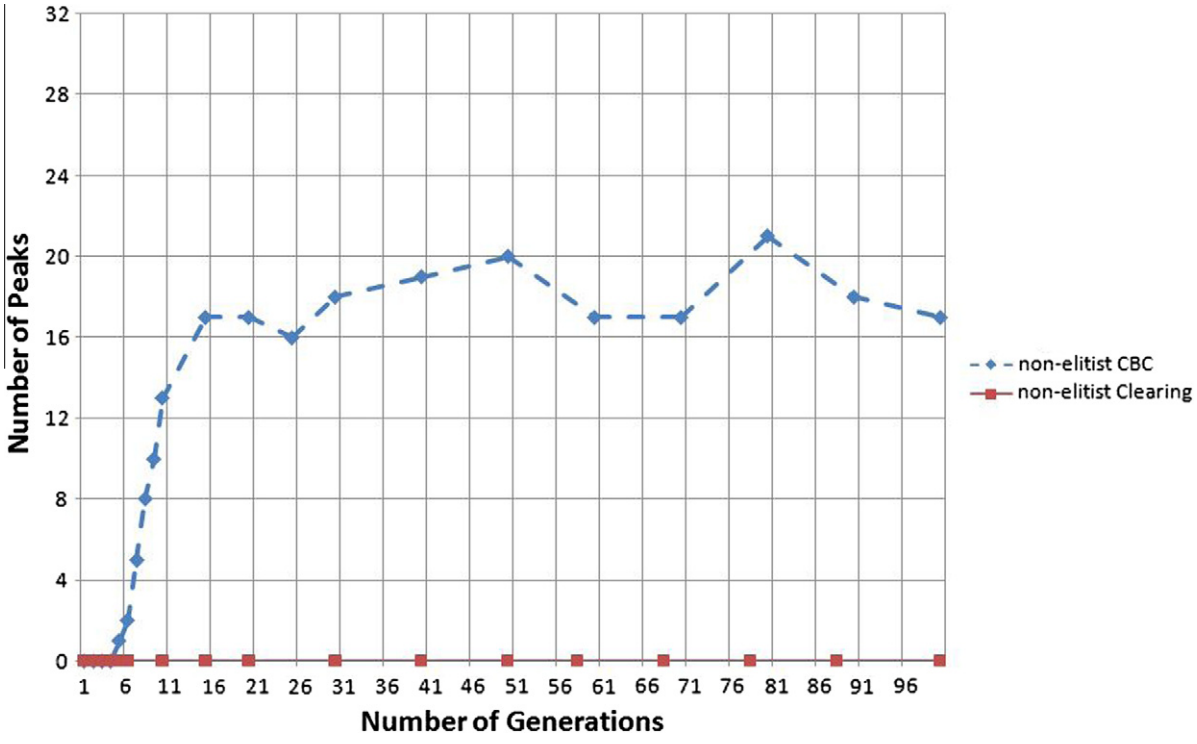


Fig. 9 Non elitist CBC vs. non elitist standard clearing results.

includes comparing the performance of both elitist and non-elitist versions of the CBC procedure.

Fig. 9 shows the average response of non-elitist versions of the CBC procedure ($SP = 10$) vs. the standard clearing procedure for solving the M7 function.

As noted from Fig. 9, the performance of non-elitist standard clearing is zero, while the performance of the CBC procedure is very high.

The CBC procedure starts finding solutions from generation 5, and also finds more than 16 optima at very early generations

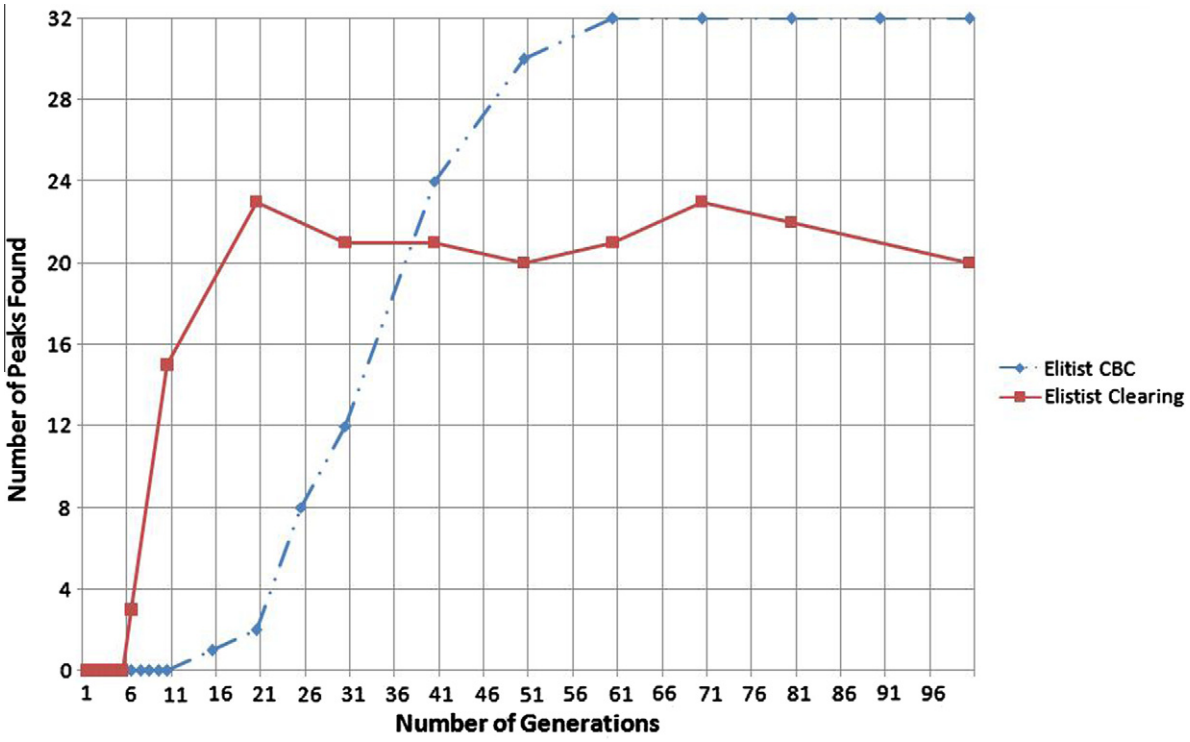


Fig. 10 Elitist CBC vs. elitist standard clearing results.

(generation number15), while in the standard clearing no optima were found.

Fig. 10 shows the average response of elitist versions of the CBC procedure ($SP = 10$) vs. the standard clearing procedure for solving M7 function.

As noted from Fig. 10, in early generations (specifically till generation 35) more than 20 optima were obtained in the case of the CBC procedure, while in the standard clearing procedure a much lower number of solutions were found. This can be explained by the fact that application of clearing in the standard clearing procedure was invalid at these early generations where heterogeneity of individuals is relatively high. This caused distraction from some local optima whose attractors were cleared away. Even when selecting more than one winner to represent a subpopulation they were not helpful as they were probably very close to each other and local optima that would have lead to global optima were still disregarded through clearing. The first solution obtained by CBC was at generation 5 while in the standard clearing procedure peaks do not start to appear before generation 15, whereas 20 peaks had already been detected by CBC.

Hence, we conclude that the CBC procedure is more efficient in finding first global optima than the standard clearing procedure since solutions appear very early using the CBC procedure. This means that if a single global maximum is targeted CBC is more efficient. However; if several optima are targeted an additional processing loss must be added to CBC. Computational complexity is also in favour of CBC, as described in next section.

Complexity

Complexity is divided into two parts. The first deals with calculating the standard deviation for all subpopulations to decide which to clear off and which not. The second deals with the case when standard deviation is above the threshold values. Then certain comparisons are necessary to select which individuals to remove and which not.

For the first part, the overall complexity for computing the standard deviation is the sum of complexities of all computations for calculating the standard deviation in all subpopulations. As given in (1), each subpopulation includes M individuals. Hence, the standard deviation of each subpopulation is calculated as follows

$$\sigma = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(x_i - \bar{x})^2} \tag{3}$$

where $\bar{x}$ is the mean of the values xi within the subpopulation, defined as:

$$\bar{x} = \frac{x_1 + x_2 + \ldots + x_M}{M} = \frac{1}{M}\sum_{i=1}^{M} x_i \tag{4}$$

Hence, for each subpopulation the complexity is $\mathbf{O}(\boldsymbol{M})$.

Now, assuming that the number of subpopulations within each iteration is $\boldsymbol{c}$, then in a single iteration, the total complexity for all $\boldsymbol{c}$ subpopulations is $\mathbf{O}(\boldsymbol{cM})$.

For the second part, we should note that when the standard deviation is less that the given threshold, all elements within the subpopulation are cleared. Now assume that $\boldsymbol{P}$ is the probability that the standard deviation is less than the given threshold. In this case no comparisons are required since all elements in the subpopulation will be cleared.

If the standard deviation is greater than the threshold, then only those individuals within "Niche Radius" from pivot are cleared. To specify the individuals to be cleared in this case, all individuals within the subpopulation must be compared

with the pivot of the subpopulation to determine their distance. This requires M more comparisons. As the probability of clearing a subpopulation is P, then the number of subpopulations that were NOT cleared is $(1 - P)$, and hence, the average number of comparisons for this second part of complexity is $(1 - P)M*c$ i.e., $\mathbf{O}(cM)$. Hence, the overall complexity of the CBC procedure for both parts is of the order $\mathbf{O}(cM)$.

Now, to compare with the standard clearing procedure [5], in a single iteration, creating a single subpopulation requires comparing its dominant individual with all the individuals that have not been assigned to a subpopulation. Hence the complexity of creating a single subpopulation is $\mathbf{O}(N)$ where N is the population size. So for b subpopulations the overall complexity is $\mathbf{O}(bN)$.

By comparing the complexity of the CBC procedure and the standard clearing procedure for a single generation the following is noted:

CBC requires an extra presorting stage for each subpopulation creation. The complexity of each is $N \log N$ (where N is population size) if a merge sort is used or $N6/5$ if a shell sort is used. Hence the total add on complexity (worst case) is $cN \log N$.[1]

On the other hand, the number of comparisons required by the CBC procedure to create subpopulations is less than the required comparisons for the clearing procedure because the CBC procedure depends on subpopulation elements only ,and not on all population elements.

Given that CBC requires fewer generations to reach the first optimum than standard clearing, so for a population size = 600[5] the total complexity will be **2.78** $(\log 600 = 2.78)N*G$ (where G is the total number of generations).

Conclusion and future work

In this paper the CBC procedure has been presented. The complexity of the standard clearing procedure and the CBC procedure is comparable.

The ability of the CBC procedure to verify the validity of clearing before applying it by checking the heterogeneity of the individuals within the subpopulation has prevented the clearing of local attractors at early stages and thus enabled it to reach solutions much earlier than standard clearing.

For applications that focus on reaching a solution as fast as possible, the CBC procedure is definitely better. As more optima are requested the subpopulation size must be decreased for the CBC, adding more processing requirements and thus decreasing its competitiveness with the standard clearing procedure. Otherwise, for applications that target all possible solutions and do not care about time, the standard clearing algorithm will be the best choice.

It is intended to modify the proposed CBC procedure to enhance its performance by modifying the method for creating subpopulations.

Also it is intended to extend the application of CBC to other real life applications in order to test its performance. The scheduling problem is targeted. Results will be published to verify the efficiency of CBC.

[1] The sorting step has the same complexity every time because the deleted individuals are only tagged not deleted from the array. Hence, the same number of individuals is sorted each time.

References

[1] Goldberg DE. Genetic algorithms in search, optimization and machine learning. 1st ed. Addison-Wesley Professional; 1989.

[2] Mahfoud SW. Niching methods extend genetic algorithms, http://citeseer.ist.psu.edu/mahfoud95niching.html; 1995.

[3] Deb K, Goldberg DE. An investigation of niche and species formation in genetic function optimization. In: Proceedings of the third international conference on genetic algorithms. George Mason University, United States: Morgan Kaufmann Publishers Inc; 1989. p. 42–50.

[4] Beasley D, Bully DR, Martinz RR. A sequential niche technique for multimodal function optimization. Evol Comput 1993;1(2):101–25.

[5] Pétrowski A. A clearing procedure as a niching method for genetic algorithms. In: Proceedings of IEEE international conference on evolutionary computation. Nagoya; 1996. p. 798–803.

[6] Cioppa AD, De Stefano C, Marcelli A. Where are the niches? Dynamic fitness sharing. IEEE Trans Evol Comput 2007;11(4):453–65.

[7] Ellabaan MMH, Ong YS. Valley-adaptive clearing scheme for multimodal optimization evolutionary search. In: 9th international conference on intelligent systems design and applications; 2009. p. 1–6.

[8] Shir OM, Bäck T. Dynamic niching in evolution strategies with covariance matrix adaptation. In: 2005 IEEE congress on evolutionary computation, IEEE CEC 2005, vol. 3; 2005. p. 2584–91.

[9] Shir OM, Bäck T. Niche radius adaptation in the CMA-ES niching algorithm. In: Shir OM, Bäck T, editors. Parallel problem solving from nature – PPSN IX. Berlin/Heidelberg: Springer; 2006. p. 142–51.

[10] Jelasity M, Dombi J. GAS, a concept on modeling species in genetic algorithms. Artif Intell 1998;99(1):1–19.

[11] Jelasity M. UEGO, an abstract niching technique for global optimization. In: Jelasity M, editor. Parallel problem solving from nature – PPSN. Berlin/Heidelberg: Springer; 1998. p. 378–87.

[12] Ortigosa PM, Garcia I, Jelasity M. Two approaches for parallelizing the UEGO algorithm. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.4.7973; 2001.

[13] Holland JH. Adaptation in natural and artificial systems. The MIT Press; 1992.

[14] Goldberg D, Deb K, Horn J. Massive multimodality, deception and genetic algorithms. In: Proceeding of the 2nd international conference of parallel problem solving from nature, vol. 2; 1992. p. 37–46.

[15] Hadhoud MM, Darwish NM, Fayek MB. A context based niching methods for niching GA (Genetic Algorithms) to solve the scheduling problem. Cairo: Faculty of Engineering; 2009.