
A New Selection Operator Dedicated to Speciation

A. Pétrowski

Département Informatique, Institut National des Télécommunications
9 rue Charles Fourier, 91011 Evry Cedex, France
E-mail: ap@etna.int-evry.fr , URL: <http://etna.int-evry.fr/~ap>

Abstract

Most niching methods create and maintain subpopulations of individuals characterized by some similarities. This paper defines the clearing procedure as a niching method that supplies the available resources of a niche only to the best individuals of each subpopulation: the winners. The clearing is naturally adapted to elitist strategies. Elitist clearing preserves good individuals from the destructive effects of genetic drift and reproduction operators, while maintaining a high level of diversity. These properties can dramatically improve the performance of genetic algorithms used for multimodal optimization. The basic clearing procedure selects the winners for reproduction. A standard selection operator then generates a competition between the winners from every subpopulation. This paper shows, through experiments, that this kind of over-selection is harmful. The concept of clearing based selection operator that ensures an equal number of offspring to every winner, regardless of its fitness is introduced. The experiments involve both easy and difficult multimodal function optimizations. They show that a clearing based selection operator can reduce the premature convergence rate, compared to a clearing procedure associated with an SUS selection.

1 GENETIC ALGORITHMS AND MULTIMODAL OPTIMIZATION

A simple genetic algorithm (Goldberg, 1989) (SGA) is suitable for searching for the optimum of unimodal functions in a bounded search space. However, both experiments and analysis show that the SGA cannot find the multiple global maxima of a multimodal function (Goldberg, 1989)(Mahfoud, 1995). This limitation can be overcome by a mechanism that creates and maintains

several subpopulations within the search space, in such a way that each highest maximum of the multimodal function can attract one of them. These mechanisms are referred to as “niching methods” (Mahfoud, 1995).

Most niching methods gather together the individuals which present some similarities into subpopulations. Regulating mechanisms are necessary to stabilize these subpopulations generation by generation. These mechanisms can be implemented within the operators that are applied at each generation of the genetic algorithm:

- ◇ the selection operator selects the individuals for the reproduction process;
- ◇ the reproduction operator modifies the selected individuals to generate new individuals. This operator can be composed of several elementary operators such as the crossover and the mutation;
- ◇ the substitution operator determines which individuals must disappear from the current population to make room for the new ones.

The sharing method (Goldberg, Richardson, 1987) is a niching mechanism which interferes with the selection operator. It is based on the sharing of limited renewal resources, between individuals characterized by some similarities, according to the principle stated by J.H. Holland in 1975 (Holland, 1992). The basic method is known for its reliability, however, it suffers from some drawbacks such as the requirement of large population sizes, and its algorithmic complexity which is $\mathcal{O}(n^2)$, where n is the size of the population. It is also necessary to know the value of parameter σ_{share} which strongly depends on the distances between the searched maxima. This latter parameter is often difficult to estimate. These drawbacks have been removed in part by the works of Goldberg et al. (1992) and Yin and Gernay (1993).

The restricted mating techniques take place within the reproduction operator. They allow the individuals to mate preferably if they belong to a same subpopulation. The membership of an individual is determined through the value of a label concatenated to its genotype. The label, also named tag bits, can mutate, allowing the individuals

to change of subpopulation: this is a migration mechanism. This kind of method does not increase the algorithmic complexity, from which derives its advantage. Furthermore it can be naturally implemented on parallel machines (Cantú-Paz, 1995). W. Spears combines this concept with the sharing method to improve its reliability (Spears, 1995). The behavior of this technique facing difficult problems is a topical question.

The crowding (De Jong, 1975) takes place within the substitution operator. According to this method, a new individual replaces the most similar individual from a sample of the population. The deterministic crowding (Mahfoud, 1995) improves the method, leading towards a better reliability. This technique does not affect the complexity of the genetic algorithm. Mahfoud showed that this method was able to solve hard problems, but it requires a great number of evaluations of the fitness function, compared to some other techniques.

The clearing procedure previously presented in (Petrowski, 1996) also derives from the niching principle stated by J.H. Holland like the sharing method. But, 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. It is applied at the selection operator level. Its reliability is similar to that of the basic sharing method with a lower complexity and much smaller population sizes.

This paper first presents the basic clearing procedure and subsequently an elitist variant. Next, some improvements are proposed to reduce the complexity and to control the subpopulations into an easier and more efficient way. The utility of a selection operator associated with a clearing procedure is then discussed and the concept of “clearing based selection operator” is introduced. Section 3 describes some tests on both easy and difficult functions. Finally, these results are compared with those obtained from sharing and deterministic crowding.

2 CLEARING THE SEARCH SPACE

2.1 PRINCIPLES

The clearing procedure is applied after evaluating the fitness of individuals and before applying the selection operator. Like the sharing method, the clearing algorithm uses a dissimilarity measure between individuals to determine if they belong to the same subpopulation, or not. This value could be the Hamming distance for binary coded genotypes, the Euclidian distance for “real coded” genotypes or it could be defined at the phenotype level.

Each subpopulation contains a dominant individual: the one that has the best fitness. The basic clearing algorithm preserves the fitness of the dominant individual and resets

the fitness of all other individuals of the same subpopulation to zero. As such, the clearing procedure fully attributes the whole resource of a niche to a single individual: the *winner*. The winner takes all resources rather than sharing them with the other individuals of the same niche, as is done in the sharing method.

A subpopulation can be defined with a niche radius similar to the one used in the basic sharing method (Goldberg, Richardson, 1987). Thus, an individual belongs to a given subpopulation if its dissimilarity with the dominant individual is less than a given threshold σ_{clear} : the *clearing radius*.

With such a mechanism, the niche of an individual is not generally known. In effect, it can be dominated by several winners. On the other hand, for a given population, the set of the winners is unique. This proposition is proved by induction: the individual that has the strongest fitness in the population is necessarily a winner. The winner and all the individuals that it dominates are then fictitiously removed from the population. We proceed in the same way with the new population which is then obtained. Thus, the list of all the winners is produced after a certain number of steps.

It is also possible to generalize the clearing algorithm by accepting several winners chosen among the best individuals of each niche. The capacity of a niche is defined as the maximum number of winners that this niche can accept. Notice that if a capacity greater than 1 is chosen, the set of winners for a given population is not generally unique. There is at least one reason to want capacities greater than 1: if the capacities are equal to the population size, the clearing effect vanishes and the search method becomes a standard GA. Thus, choosing capacities between one and the population size offers intermediate situations between the maximum clearing and a standard GA.

2.2 THE CLEARING PROCEDURE

A plain version of the clearing procedure is presented below in pseudo code. \mathbf{P} and n are global variables. “*Sigma*” is the clearing radius and “*Kappa*” is the capacity of each niche. “*nbWinners*” indicates the number of winners of the subpopulation associated with the current niche. Population \mathbf{P} can be considered as an array of n individuals.

The plain algorithm uses three functions:

- ◇ **SortFitness(\mathbf{P})** sorts population \mathbf{P} according to the fitness of the individuals by decreasing values. The whole population is ranked for the sake of clarity in this version of the algorithm. A more optimized algorithm would only sort the dominant individuals.
- ◇ **Fitness($\mathbf{P}[i]$)** returns a reference on the fitness of the i -th individual of population \mathbf{P} .

◇ **Distance(P[i], P[j])** returns the distance between two individuals i and j of population P .

```
function Clearing(Sigma, Kappa)
begin
  SortFitness(P)
  for i = 0 to n - 1
    if Fitness(P[i]) > 0
      nbWinners := 1
      for j = i + 1 to n - 1
        if Fitness(P[j]) > 0 and
           Distance(P[i], P[j]) < Sigma
          if nbWinners < Kappa
            nbWinners := nbWinners + 1
          else
            Fitness(P[j]) := 0.0
          endif
        endif
      endfor
    endif
  endfor
end
```

2.3 AN ELITIST STRATEGY FOR THE CLEARING PROCEDURE

An elitist strategy preserves the good individuals from the destructive effects of the reproduction operators and the genetic drift. Such a strategy memorizes the best individual(s) of a population found before the application of genetic operators and passes it (them) unaltered on to the next generation. K.A. De Jong noted that this strategy can improve the performance of a GA for a unimodal fitness function, but also that the performance is degraded for the “F5” function, for example, which is multimodal (De Jong, 1975). An important reason of this low performance lies in the increased premature convergence hazards during which a large number of individuals concentrate on some maxima of the fitness function.

One way to avoid this problem is to limit individual density at every region of the search space. This is precisely one of the effects of niching methods. Then, the problem is to determine the best individuals of every subpopulation in order to preserve them. Now, the clearing procedure supplies these individuals naturally: they are the winners. If the preservation of all the winners immobilizes too great a fraction of the population to achieve good convergence properties, it is possible to use a more restrictive choice criterion, such as, for example, preserving only the winners with a fitness greater than the average before clearing. This is the method chosen for the experiments described below. Another possibility is to memorize only the dominant individual of each subpopulation. It has been shown in (Petrowski, 1996) that such an elitist strategy associated with a clearing procedure can dramatically improve the performance.

2.4 IMPROVING THE CLEARING PROCEDURE

2.4.1 Niche capacities

A great niche capacity concentrates many individuals in the higher peaks. This reduces the number of subpopulations and saves computation time. However, the premature convergence rate increases due to the loss of diversity. This has been confirmed by experiments detailed below. This drawback must be avoided. Therefore a niche capacity set to 1 is required for the best results.

2.4.2 Complexity

In (Petrowski, 1996), an upper bound of the complexity for the basic clearing procedure was established as $\mathcal{O}(cn)$, where c is the number of subpopulations and n is the population size. This complexity is equal to that of some improved sharing methods. However, c could be of the order of n if the number of peaks was greater than the size of the population. The same phenomenon happens if the clearing radius is chosen too small. In these cases, the complexity is identical to the basic sharing method complexity, i.e. $\mathcal{O}(n^2)$. One way to reduce this former one is to build subpopulations by a hierarchical clustering method (Petrowski, 1997). Then, the complexity becomes $\mathcal{O}(n \log n)$.

2.4.3 Is the clearing radius required ?

The correct estimate of the clearing radius σ used in the basic method is a difficult problem. However, the clearing radius is only a simple way to define subpopulations. It is also similar to the sharing radius. Thus, both methods (Petrowski, 1996) can be compared quite simply. Now, subpopulations can be built according to the fitness landscape using the above mentioned clustering algorithm. The clustering algorithm should then automatically redistribute the population into subpopulations, so that each subpopulation is associated with one peak. This is performed with fast local studies of the fitness landscape in the neighborhoods of the dominant individuals (Petrowski, 1997).

2.5 A CLEARING BASED SELECTION OPERATOR

Is a selection operator useful with the clearing procedure ?

Up to now, the clearing was considered as a niching method similar to the sharing method. In a standard way, such a procedure must be associated with a selection operator to enable the evolutionary algorithm to work.

Concerning the clearing procedure, it can be noticed that the non-winner individuals of every subpopulation disappear from the population at the next generation because their fitnesses are reset to zero. Thus, all this happens as if the clearing procedure performs a preselection applied before the actual selection operator.

A standard selection operator favours the reproduction of individuals possessing a high fitness to the detriment of the others. Associated with the clearing procedure, the selection operator acts as if the subpopulations were in competition. Consequently, this operator destroys the subpopulations which have the lowest fitnesses. The result is a loss of diversity within the global population and this could imply an increasing rate of premature convergence. On the other hand, the selection operator guarantees that the subpopulations with the highest fitnesses have the largest population sizes. Thus, when there is no premature convergence, the search for all the global maxima should be more efficient.

The suppression of the selection operator is equivalent to giving an equal number of offspring to every winner, regardless of their fitnesses. This reduces the selection pressure. Consequently, this suppression should imply a lower premature convergence rate, while the maxima should be found with a greater number of generations.

The experiments have confirmed these hypotheses for 4 multimodal functions which are characterized by different properties and unequal hardness: "M6", "Roots", "M7" and "M9". Subsequently, the term "*clearing based selection operator*" defines a clearing procedure, associated with a mechanism which ensures that all winners have an equal expected number of offspring.

3 EXPERIMENTS

The experiments consist of comparisons between the clearing procedure associated with an SUS operator (Baker, 1987) and the clearing based selection operator. They have two aims:

- ◇ First of all, the variation of the premature convergence rates according to the type of selection operator can be estimated at least for the functions under consideration.
- ◇ Second, they give an estimate of the mean number of fitness evaluations needed for the convergence depending on the type of selection operator.

The 4 functions "M6", "Roots", "M7" and "M9" have been processed with a clearing procedure speeded up by a hierarchical clustering (Petrowski, 1997). This kind of clustering algorithm adapts itself to the function landscape and avoids setting up parameters difficult to estimate such as the clearing radius. For all the functions, the following parameters are set:

- ◇ Binary coding of the genotype;
- ◇ Single point crossover, the crossover rate is set to 1;
- ◇ The mutation rate is zero. Some diversity is brought in the population by replacing 10% of its individuals by random individuals for each generation;
- ◇ The minimal number of subpopulations built by the clustering algorithm is set to 16 in order to keep a large enough diversity;
- ◇ Capacity κ of the niches is chosen equal to one;
- ◇ The adaptive elitist strategy described above (section 2.3) is used.

These parameters have been intentionally set identical for all experiments, in spite the variety of the functions, to show that these choices are not critical and that this kind of evolutionary algorithm is robust. The easy functions "Roots" and "M6" have been tested with insufficient population size in order to clearly show premature convergence phenomena. The tests with more difficult functions "M7" and "M9" have been performed with adequate population sizes. They aimed at showing that replacing the SUS by the clearing based selection operator does not affect the quality of the convergence. These tests even showed a greater efficiency for the clearing based selection operator.

Each series of tests related to a given function yields two curves: one for the clearing procedure associated with the selection operator SUS and the other for the clearing based selection operator. Each curve gives the mean number of peaks found for a series of 100 successive tests vs. the number of generations. The grey areas represent the confidence interval with an error probability of 1%.

3.1 THE "M6" FUNCTION

M6 (Mahfoud, 1995) is derived from the "F5" function used by De Jong in his dissertation (De Jong, 1975). It shows 25 maxima located as though on a 2-dimensional array with values ranging from 476.191 to 499.002. The purpose of the present experiments was not to find only the global maximum located at (-32, -32), but to locate all the maxima.

M6 is defined as follows:

$$M6(x, y) = 500 - \frac{1}{0.002 + \sum_{i=0}^{24} \frac{1}{1+i+(x-X_i)^6+(y-Y_i)^6}}$$

where (X_i, Y_i) is the location of the i -th maximum :

$(X_i, Y_i) \in \{$
 $(-32, -32), (-32, -16), (-32, 0), (-32, 16), (-32, 32),$
 $(-16, -32), (-16, -16), (-16, 0), (-16, 16), (-16, 32),$
 $(0, -32), (0, -16), (0, 0), (0, 16), (0, 32),$
 $(16, -32), (16, -16), (16, 0), (16, 16), (16, 32),$
 $(32, -32), (32, -16), (32, 0), (32, 16), (32, 32) \}$

In the search space, $x \in [-64, 64]$, and $y \in [-64, 64]$. x and y are each encoded on 20 bits. Thus, the genotype has a length of 40 bits.

All the peaks are always found over 100 successive tests with a population size of 100 individuals. But, it was not possible to clearly distinguish the difference of convergence behavior between a clearing based selection operator and a clearing procedure associated with an SUS. So, the reported tests are related to an insufficient population size of 50 individuals, in order to increase the premature convergence rate. Figure 1 represents the curves that give the mean number of peaks found vs. the number of generations with or without the SUS operator.

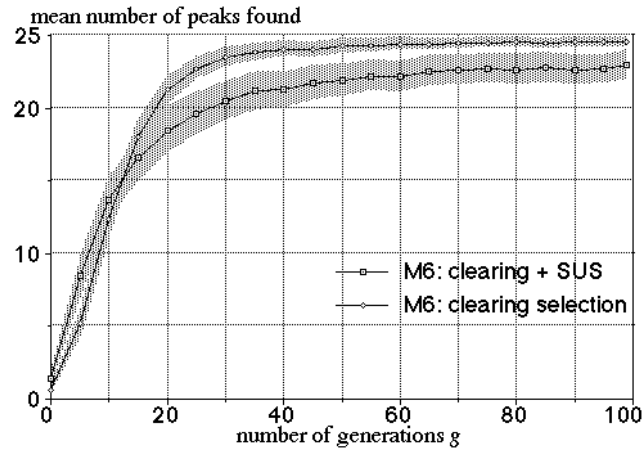


Figure 1: "M6" Function, 50 Individuals in the Population.

3.2 THE "ROOTS" FUNCTION

The "Roots" function is defined as follows:

$$\text{Roots}(z) = \frac{1}{1 + |z^6 - 1|} \quad \text{where } z \in \mathbf{C}.$$

This function takes their maxima at the sixth roots of unity in the complex plane.

It presents a large plateau at the height of 0.5 centered at $(0, 0)$ surrounded by six thin peaks at the height of 1.0 (figure 2). The maxima are quite easy to find but the experiments lead one to think that the problem seems to be harder than "M6".

Let x and y be the real numbers such as $z = x + iy$. In the search space, $x \in [-2, 2]$, and $y \in [-2, 2]$. x and y are each

encoded on 20 bits. Thus, the genotype has a length of 40 bits.

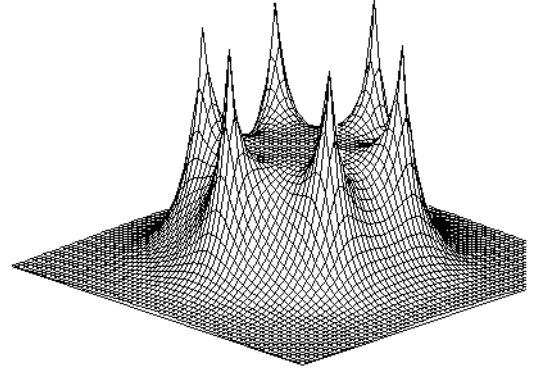


Figure 2: the "Roots" Function

As for "M6" function, all the peaks are found over 100 successive tests with a population size of 100 individuals. Figure 3 is related to an insufficient population size of 50 individuals, in order to clearly show the effect of the SUS operator on the premature convergence rate.

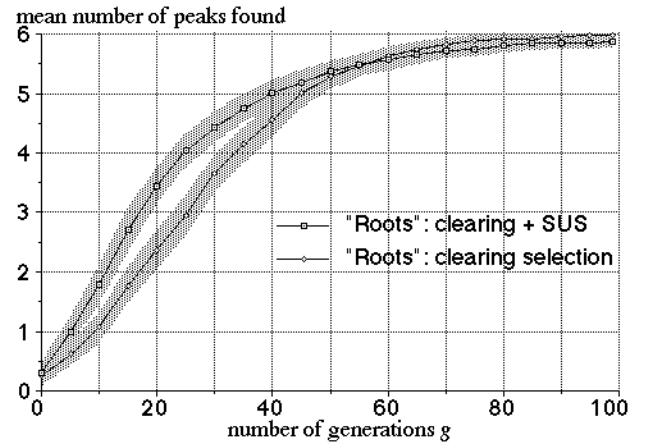


Figure 3: "Roots" Function, 50 Individuals in the Population.

3.3 THE "M7" FUNCTION

M7 is the name given by Mahfoud to a massively multimodal deceptive function previously presented in (Goldberg *et al.*, 1992). **M7** is defined as follows:

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

where $\forall k, x_k \in \{0, 1\}$. Function $\mathbf{u}(x)$ is defined for the integer values 0 to 6 (figure 4). 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 \mathbf{u} has been specially built to be deceptive.

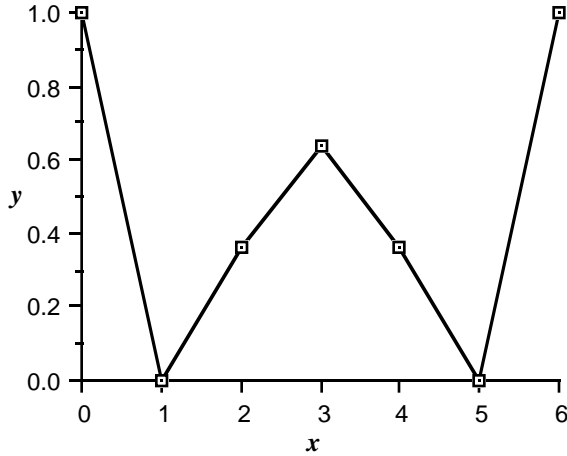


Figure 4: $y = \mathbf{u}(x)$

Function **M7** has 32 global maxima of value equal to 5, and several million local maxima, the values of which are between 3.203 and 4.641.

The concatenation of bits x_k , directly constitute a 30 bit genotype.

Some experiments with “M7” are also presented in (Mahfoud, 1995).

All the 32 peaks are always found over 100 successive tests with a population size of 800 individuals. Figure 5 shows the results.

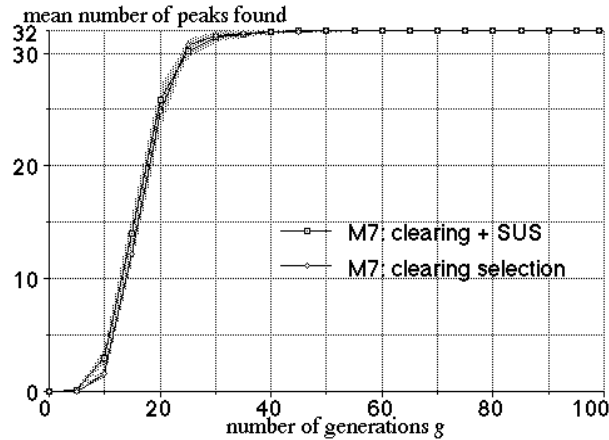


Figure 5: “M7” Function, 800 Individuals in the Population.

3.4 THE “M9” FUNCTION

M9 is the name given by Mahfoud to a function previously presented in (Horn, Goldberg, 1995). **M9** is the sum of three elementary subfunctions f_{mdG} defined for 8-bit binary vectors:

$$M9(x_0, \dots, x_{23}) = \sum_{i=0}^2 f_{\text{mdG}}(x_{8i}, \dots, x_{8i+7})$$

where $x_i \in \{0, 1\}$

Let \mathbf{G} be the set of the maxima of f_{mdG} . Let s be a 8-bit binary vector.

$$f_{\text{mdG}}(s) = \begin{cases} 10 & \text{if } s \in \mathbf{G} \\ \min_{g \in \mathbf{G}} H(s, g) & \text{otherwise} \end{cases}$$

where $H(s, g)$ is the Hamming distance between s and g . \mathbf{G} contains three arbitrarily chosen points: 00000000, 10001100, 01001010.

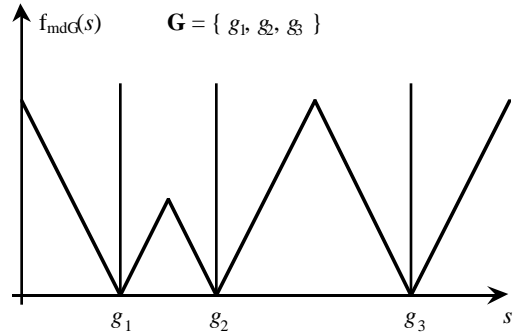


Figure 6: $f_{\text{mdG}}(s)$ Function in the Monodimensional Real Case.

“M9” is difficult because the peaks are isolated and located at the lower points of large basins (Horn, Goldberg, 1995). To illustrate this, one can consider a generalization of the $f_{\text{mdG}}(s)$ function to real numbers by replacing the Hamming distance by the Euclidian distance. Figure 6 gives a representation of such a function in the monodimensional case.

All the peaks are always found over 100 successive tests with a population size of 1000 individuals. Figure 7 shows the results.

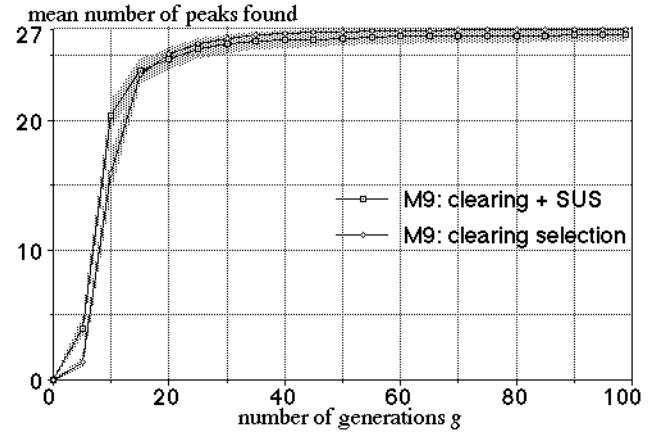


Figure 7: “M9” Function, 1000 Individuals in the Population

Other experiments with “M9” are also presented in (Mahfoud, 1995).

3.5 COMPARISONS

To complete these results, the table below gives the mean number of fitness evaluations needed to find all the maxima of the processed functions. The population sizes are indicated for each of them. It can be noticed that the number of evaluations required for these functions are equally important without giving an advantage to the clearing operator or to the SUS operator.

number of eval.	clearing + SUS	clearing based selection
"M6" ($n = 100$)	$1.2 \cdot 10^3$	$1.0 \cdot 10^3$
"Roots" ($n = 100$)	$1.4 \cdot 10^3$	$1.7 \cdot 10^3$
"M7" ($n = 800$)	$1.2 \cdot 10^4$	$1.2 \cdot 10^4$
"M9" ($n = 1000$)	$1.4 \cdot 10^4$	$1.4 \cdot 10^4$

These results for “M6”, “M7” and “M9” might be compared with sharing and deterministic crowding from the work of S.W. Mahfoud (1995). The GA used was associated with a hill-climbing to speed up the convergence. The table below gives the total number of evaluations required by the GA and the hill-climbing. The tests have been performed by increasing the population size until the convergence is achieved. The number of evaluations was upper bounded by $1.5 \cdot 10^6$. The population sizes were not given in this work. This table shows that these numbers are much greater than those required by the GA associated with the clearing operator.

number of eval.	sharing method	deterministic crowding
"M6"	$1.3 \cdot 10^4$	$> 1.5 \cdot 10^6$
"M7"	$> 1.5 \cdot 10^6$	$1.0 \cdot 10^5$
"M9"	$> 1.5 \cdot 10^6$	$1.3 \cdot 10^6$

More thorough comparisons can be found in (Petrowski, 1996).

4 CONCLUSION

The clearing operator is an efficient selection operator dedicated to speciation. It creates a maximal selection pressure inside subpopulations as only the winners can survive. And it annihilates the selection pressure between

subpopulations. The clearing based selection operator has the following properties:

- ◊ It is based on the same niching principle as the sharing method.
- ◊ The time complexity of the clearing is lower than that of the sharing method associated with an SUS. It could be $n \log n$ with clustering methods such those described in (Petrowski, 1997).
- ◊ The clearing is directly compatible with elitist strategies. This improves convergence in great proportions by preventing the genetic drift and reproduction operators from destroying good individuals.

A clearing procedure used without a standard proportionate selection operator offers a reduced premature convergence rate for all the functions considered in this paper, although the number of evaluations is of the same magnitude for a clearing associated with an SUS. Thus, the clearing based selection operator behaves as a selection operator and seems to work more efficiently.

Such a selection operator generates a selection noise that is not of the same nature as that of a proportionate selection operator. This noise comes from the incorrect distributions of the population that does not separate two peaks or more. This happens when the clearing radius is set too wide by the user, or when the number of individuals in the neighborhood of each peak is too small to determine the subpopulations in a reliable way with a clustering technique. (Petrowski, 1997). In this last case, a large enough population size is required to reduce this noise to an acceptable value. Additional research is needed to study this problem.

This paper has presented the successful solving of both easy and difficult multimodal problems using the clearing based selection operator in a very efficient way when compared with the performance of other kinds of niching methods.

Acknowledgments

The author wishes to thank *K. English* and *C. Delassus* for reading the manuscript. He is also grateful to the anonymous referees for their advice in improving it. This work was supported by *France Telecom*.

References

J.E. Baker, “Reducing Bias and Inefficiency in the Selection Algorithm”, in *Genetic Algorithms and their Applications: Proc. of the 2nd Int. Conf. on Genetic Algorithms*, pp. 14-21, 1987

- E. Cantú-Paz, *A summary of Research on Parallel Genetic Algorithms*, IlliGAL report No 95007, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign, July 1995
- K.A. De Jong, *An Analysis of the Behavior of a Class of Genetic Adaptive Systems*, doctoral dissertation, University of Michigan, 1975
- D.E. Goldberg, *Genetic Algorithms in Search, Optimization & Machine Learning*, Addison Wesley, 1989.
- D.E. Goldberg, K. Deb, J. Horn, "Massive Multimodality, Deception and Genetic algorithms", in R. Männer, B.Manderick (Eds.) *Parallel Problem Solving from Nature 2*, pp. 37-46, North Holland, 1992
- D.E. Goldberg, J. Richardson, "Genetic Algorithms with Sharing for Multimodal Function Optimization", in *Genetic Algorithms and their Applications: Proc. of the 2nd Int. Conf. on Genetic Algorithms*, pp. 41-49, 1987
- J.H. Holland, *Adaptation in Natural and Artificial Systems*, The MIT Press, 1992 (1st ed.: 1975)
- J. Horn, D.E. Goldberg, Genetic Algorithms Difficulty and the Modality of Fitness Landscapes, In L.D. Whitley & M.D. Vose (Eds.), *Foundations of Genetic Algorithms 3 (FOGA 3)*, pp. 243-269, Morgan Kaufmann, 1995.
- S.W. Mahfoud, *Niching Methods for Genetic Algorithms*, doctoral dissertation, University of Illinois at Urbana-Champaign, 1995.
- A. Petrowski, "A Clearing Procedure as a Niching Method for Genetic Algorithms", *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation*, pp. 798-803, IEEE, 1996
- A. Petrowski, *An Efficient Hierarchical Clustering Technique for some Niching Methods*, technical report, Institut National des Télécommunications, Evry, France, 1997
- W. Spears, "Simple Subpopulation Schemes", *Proceedings of the 3rd Annual Conference on Evolutionary Programming*, pp. 296-307, World Scientific, 1994
- X. Yin, N. Gernay, "A Fast Genetic Algorithm with Sharing Scheme Using Cluster Analysis Methods in Multimodal Function Optimization", in R.F. Albrecht, C.R. Reeves & N.C. Steele (Eds.), *Proceedings of the International Conference on Artificial Neural nets and Genetic Algorithms*, pp. 450-457, Berlin, Springer Verlag, 1993