An Investigation of Niche and Species Formation in Genetic Function Optimization

Kalyanmoy Deb and **David E. Goldberg**
Department of Engineering Mechanics
The University of Alabama
Tuscaloosa, AL 35487

Abstract

In artificial genetic search, multimodal functions are optimized by inducing the natural concepts of niche and species into a population of strings. In this paper, a number of methods are suggested for this purpose. Specifically, *crowding* and *sharing function* methods are compared on the basis of their performance on a number of test functions. Simulation results show that a GA with sharing is able to converge and distribute trials at all the peaks of the functions, whereas a GA with crowding is unable to maintain subpopulations at all the peaks. Two forms of sharing functions are considered, so-called *phenotypic* and *genotypic* sharing, and a mating restriction scheme is implemented to improve on-line performance.

1 Introduction

Over the years, genetic algorithms (GAs) have proved useful in a variety of search and optimization problems (Goldberg, 1989). As the usage of GAs has grown, objections to their performance on specific problems have arisen, and when this happens, natural remedies are often tried. For example, to achieve better performance on nonstationary functions, dominance and diploidy have been added (Goldberg and Smith, 1987); to overcome the limitations in fixed codings, inversion and reordering operators have been suggested (Goldberg, 1989). In dealing with multimodal functions, simple GAs converge to a single peak (Goldberg and Richardson, 1987), even though multiple peaks of equal quality may exist. Faced with a similar problem, nature forms stable subpopulations of organisms surrounding separate niches by forcing similar individuals to share the available resources. In artificial genetic search, a number of modifications have been tried on a simple GA to implement an analogous form of sharing (Deb, 1989; Goldberg, 1989). Among them, the methods used for the optimization of single-objective, multimodal functions are the crowding method suggested by De Jong (1975) and two sharing schemes suggested by Goldberg and Richardson (1987). This paper compares these schemes based on their performance on a number of single and multiparameter, multimodal functions and implements a mating restriction scheme. In the optimization of multimodal functions using sharing and mating restriction, the former helps maintain subpopulations at multiple peaks by dividing the population into different niches, while the latter helps improve the overall performance by promoting speciation among the members of each niche.

In the remainder of this paper, a brief review of a number of crowding and sharing schemes is made. Methods to calculate the sharing parameters are formulated, and experimental results comparing the performance of several GAs on a number of multimodal functions are then presented. Genotypic sharing results are discussed in the light of a simplified convergence analysis. Finally, a phenotypic mating restriction scheme is implemented together with phenotypic sharing and applied to a test function to show improvement over the on-line performance of sharing alone.

2 Niche-formation Methods

In the optimization of multimodal functions, a simple GA cannot maintain controlled competition among the competing schemata corresponding to different peaks, and the stochastic error associated with the genetic operators causes the population to converge to one alternative or other. This problem with finite populations is known as *genetic drift* (De Jong, 1975; Goldberg and Segrest, 1987). Moreover, in dealing with multimodal functions with peaks of unequal value, a simple GA converges to the best peak; whereas, in addition to wanting to know the best solution, one may be interested in knowing the location of other optima. To overcome these limitations a natural remedy is tried.

In nature, a niche is viewed as an organism's task in the environment, and a species is a collection of organisms with similar features. The subdivision of environment on the basis of an organism's role reduces inter-species competition for environmental resources, and this reduction in competition helps stable subpopulations to form around different niches in the environment. A number of methods are suggested to introduce this notion in genetic algorithms. Specifically, crowding and sharing are briefly discussed in the following.

2.1 Crowding Scheme

In De Jong's crowding (1975), separate niches are created by replacing existing strings according to their similarity with other strings in an overlapping population. Two parameters, generation gap (G) and crowding factor (CF), are defined for this purpose. Generation gap G dictates the use of an overlapping population model in which only a proportion G of the population is permitted to reproduce each generation. To induce niche in the population, the following approach is adopted. When selecting an individual to die, CF individuals are picked at random from the population, and the one which is most similar to the new individual is chosen to be replaced, where similarity is defined in terms of the number of matching alleles. The new individual (chosen by usual selection methods) then replaces this chosen individual in the population. De Jong used this scheme successfully with crowding factor $CF = 2$ and 3 and with generation gap $G = 0.1$. De Jong's crowding scheme has been subsequently used in a machine learning application (Goldberg, 1983).

2.2 Sharing Scheme

Goldberg and Richardson (1987) used Holland's (1975) sharing concept by dividing the population in different subpopulations according to the similarity of the individuals in two possible solution spaces: the decoded parameter space and the gene space. They defined a sharing parameter σ_{share} to control the extent of sharing, and they defined a power-law sharing function $Sh(d)$ as a function of the distance-metric (d) between two individuals as follows:

$$Sh(d) = \begin{cases} 1 - \left(\frac{d}{\sigma_{\text{share}}}\right)^{\alpha}, & \text{if } d < \sigma_{\text{share}}; \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

To implement the idea of sharing, an individuals's payoff is degraded due to the presence of other individuals in its neighborhood. When the proximity of the individuals is defined in the decoded parameter space, it is called *phenotypic sharing*. In two multimodal functions, Goldberg and Richardson were able to show the successful clustering of trials at the peaks. In the same study, they suggested, but did not simulate, the use of sharing based on genetic proximity or *genotypic sharing*. As they suggested, the genetic closeness of two individuals may be taken as the number of different alleles in their chromosomes (the Hamming distance between the strings).

The working of the sharing principle mainly depends on the parameter σ_{share} and must be set carefully. The parameter σ_{share} is the maximum distance between the strings necessary to form as many niches as there are peaks in the solution space. Therefore, the parameter σ_{share} depends on the number of peaks and the upper and lower bounds of the solution space. The next section discusses a method to calculate the parameter σ_{share} in both phenotypic and genotypic sharing.

3 Calculation of the Parameter σ_{share}

Since the parameter σ_{share} in phenotypic and genotypic sharing is defined in different solution spaces, the calculation of the parameter σ_{share} in each case is also different and is discussed in the following.

3.1 Phenotypic Sharing

The distance metric (d_{ij}) considered in phenotypic sharing is the distance between strings in the decoded parameter space. For a single parameter function, this may be calculated as the absolute difference of the decoded parameter values of the strings. In general, for a p-parameter function, the distance d_{ij} may be calculated using any suitable *distance-norm* in the p-dimensional space. For simplicity, the Euclidian distance in p-dimensional space is adopted here. Therefore, for the individuals $\mathrm{x}_i = [\mathrm{x}_{1,i}, \mathrm{x}_{2,i}, \ldots \mathrm{x}_{p,i}]$ and $\mathrm{x}_j = [\mathrm{x}_{1,j}, \mathrm{x}_{2,j}, \ldots \mathrm{x}_{p,j}]$ the metric d_{ij} may be calculated as

$$d_{ij} = \sqrt{\sum_{k=1}^{p} (\mathrm{x}_{k,i} - \mathrm{x}_{k,j})^2}, \quad (2)$$

where the $\mathrm{x}_{1,i}, \mathrm{x}_{2,i}, \ldots \mathrm{x}_{p,i}$ are the decoded parameters.

To estimate the parameter σ_{share}, imagine that each niche is enclosed in a p-dimensional hypersphere of radius σ_{share} such that each sphere encloses $\frac{1}{q}$ of the volume of the space, where q is the number of peaks in the solution space. The radius of a hypersphere containing the entire space is calculated as $r = \frac{1}{2}\sqrt{\sum_{k=1}^{p}(\mathrm{x}_{k,max} - \mathrm{x}_{k,min})^2}$, and the volume is calculated as $V = cr^p$ with c a constant. Dividing this volume in q parts and recognizing that the hypervolume has the same form regardless of size, σ_{share} may be calculated as follows:

$$\begin{aligned} c\sigma^p_{\text{share}} &= \frac{1}{q}\, cr^p; \\ \sigma_{\text{share}} &= \frac{r}{\sqrt[p]{q}}; \end{aligned}$$

$$= \frac{\sqrt{\sum_{k=1}^{p}(\mathrm{x}_{k,max}-\mathrm{x}_{k,min})^2}}{2\sqrt[p]{q}}. \quad (3)$$

For a single parameter function ($p = 1$), the above equation reduces to

$$\sigma_{\text{share}} = \frac{\mathrm{x}_{max}-\mathrm{x}_{min}}{2q}. \quad (4)$$

3.2 Genotypic Sharing

In genotypic sharing, the metric d_{ij} is defined as the Hamming distance between the strings, and the parameter σ_{share} is defined as the maximum number of different bits allowed between the strings to form separate niches in the population. An estimate of the parameter σ_{share} may be calculated as follows.

Consider that two binary strings s_i and s_j are of length ℓ. Comparing string s_i with string s_j, if only one bit difference is allowed between the strings, there are a total $\binom{\ell}{1}$ such strings possible. Similarly, if two distinct bits of difference are allowed, there are $\binom{\ell}{2}$ such strings possible. In general, for k distinct allowable bits of difference, there are total $\binom{\ell}{k}$ strings possible in the solution space. Since there are at most 2^ℓ strings in the solution space, it may be argued that, in general, $\binom{\ell}{k}/2^\ell$ of the whole solution space are all k-bits away from each other. A similar estimate with the strings having *at most* a k-bit difference in their gene space may then be calculated by summing all such quantities varying from zero to k (or, $\sum_{i=0}^{k}\binom{\ell}{i}/2^\ell$). On the other hand, for a q-peaked function, there are q niches in the solution space, and assuming uniform niche placement, each niche must correspond to an average of $\frac{1}{q}$ of the total solution space. Therefore, if k (or σ_{share}) is the maximum bits of difference allowed between the strings to make q-subspaces in the solution space, then

$$\frac{1}{2^\ell}\cdot\sum_{i=0}^{k}\binom{\ell}{i} = \frac{1}{q}. \quad (5)$$

For specific values of the number of peaks, q, and the string length, ℓ, the parameter k (or σ_{share}) may be estimated from the above equation. It is, interesting to note that the left hand side of this equation corresponds to a cumulative binomial distribution with probability of occurrence equal to 0.5. Therefore, a cumulative binomial distribution chart may be used for a quick calculation of genotypic σ_{share} using the above equation.

If the string length ℓ is large, the calculation of σ_{share} from the above equation becomes cumbersome; moreover, there is no cumulative binomial distribution chart available for large values of ℓ. In such cases, the parameter σ_{share} may be calculated by considering the normal approximation to the binomial distribution. Assume that the number of bits of difference over the string length ℓ between two strings is normally distributed with mean $\mu = p\ell = \frac{\ell}{2}$, and variance $\sigma^2 = \ell p(1-p) = \frac{\ell}{4}$. Considering that z^* is the required normalized bit difference corresponding to $\frac{1}{q}$ of total probability space, one may write

$$\begin{aligned}\frac{\sigma_{\text{share}}-\mu}{\sigma} &= z^*;\\ \sigma_{\text{share}} &= \frac{1}{2}(\ell+z^*\sqrt{\ell}). \quad (6)\end{aligned}$$

Therefore, a z^* corresponding to the fraction $\frac{1}{q}$ may be found from a cumulative normal distribution chart and the parameter σ_{share} may be calculated. For large values of population size and string length, the parameter σ_{share} calculated from the above equation compares well with that calculated from Equation 5.

4 Crowding versus Sharing

The crowding and the sharing methods are implemented on genetic algorithms and applied to a number of test functions to compare the performance of each scheme based on three performance criteria discussed below. A number of other test functions are considered elsewhere (Deb, 1989). This paper considers the following three functions:

$F1 : f_1(x) = \sin^6(5\pi x)$

This function has five peaks of same size in the range $0 \le x \le 1$.

$F2 : f_2(x) = e^{-2\ln 2(\frac{x-0.1}{0.8})^2}\sin^6(5\pi x)$

This function has five peaks of unequal size in the range $0 \le x \le 1$.

Similar functions were used in Goldberg and Richardson (1987). They are considered here to compare different schemes using new performance criteria.

$F3$: Himmelblau's Function

$$f_3(x_1,x_2) = (x_1^2+x_2-11)^2+(x_1+x_2^2-7)^2.$$

This function has four optima in the region $-6.0 \le x_1, x_2 \le 6.0$, where x_1 and x_2 are two parameters (Reklaitis, Ravindran, and Ragsdell, 1983). The optimum points correspond to the minima of the above function. Therefore, to use this function in genetic search technique, the function is suitably transformed into a maximization problem.

The performance of the algorithms on these functions are judged on the basis of two criteria: the convergence and the distribution of trials at all the peaks. De Jong's (1975) on-line and off-line performance measures are used to measure the best and average convergence of trials at the peaks. In the case of multimodal functions, simple on-line or off-line performance measures do not judge the distribution pattern of the trials over the peaks; moreover, a high on-line or off-line performance measure is not meaningful when the function has unequal peaks. Therefore, simple on-line and off-line performance statistics are insufficient to judge the performance of a GA in the case of multimodal functions. To characterize the distribution of trials over the peaks, a chi-square-like criterion is developed, where the actual distribution is compared to an ideal distribution.

An ideal distribution pattern of trials over the peaks may be calculated from the modified k-armed bandit problem introduced by Holland (1975). In a modified k-armed bandit problem, the individuals queued up behind each arm correspond to a niche and if the individuals of one arm are not allowed to share with the individuals of other, they form a stable subpopulation of individuals in each arm in proportion to their arm payoff values; however, in a real-world function such clear definitions of peaks and the number of individuals associated with peaks are not possible. Therefore, trials having fitness values within a fraction ϵ of the representative peak fitness value are associated with that peak. The solution space then consists of q subspaces corresponding to each peak so defined and one other subspace that includes the individuals not representing any peak. To estimate a chi-square-like deviation measure, the expected value and the corresponding variance of the number of individuals representing each subspace are required. For the on-peak individuals, these parameters may be calculated from the peak-fitness values, f_i, and the population size, N, as $\mu_i = \frac{f_i}{\sum f_i} \times N$ and $\sigma_i^2 = Np_i(1 - p_i)$, where $p_i = \frac{\mu_i}{N}$. But, the expected number and the variance of the individuals that do not represent any peak (μ_{q+1} and σ_{q+1}^2) cannot be calculated in this fashion since there is no single representative fitness value for these individuals. Nonetheless, these parameters may be estimated by setting the sum of the individuals in all $(q+1)$ subspaces equal to the population size, N.

Let $X_1, X_2, \ldots X_q$ represent the number of individual in the successive peaks and let X_{q+1} denote the number of individuals that do not represent any peak. The variables $X_1, X_2, \ldots X_q$ are assumed to be independent of each other. Another variable X_T is chosen such that $X_T = \sum_{i=1}^q X_i$.

The expected value of the variable X_T may then be written as $E[X_T] = \sum_{i=1}^q E[X_i] = \sum_{i=1}^q \mu_i = N$, and similarly, $\sigma_T^2 = \sum_{i=1}^q \sigma_i^2$. Recognizing that,

$$\begin{aligned} X_{q+1} &= N - X_T; \\ \mu_{q+1} &= 0; \qquad (7) \\ \text{and } \sigma_{q+1}^2 &= \sigma_T^2; \end{aligned}$$

$$\begin{aligned} &= \sum_{i=1}^{q} \sigma_i^2; \\ &= \sum_{i=1}^{q} \mu_i \left(1 - \frac{\mu_i}{N}\right). \qquad (8) \end{aligned}$$

Thereafter, a chi-square-like distribution error may be defined as

$$\text{performance measure} = \sqrt{\sum_{i=1}^{q+1} \left(\frac{X_i - \mu_i}{\sigma_i}\right)^2} \qquad (9)$$

This measure estimates the deviation of the actual distribution of individuals X_i from the ideal distribution μ_i in all the $q+1$ subspaces where the smaller the measure, the better the method.

4.1 Simulation Results

The GA parameters used in the simulations are as follows:

maximum generation	:	200
population size	:	100
string length (binary coded)	:	30
probability of crossover	:	0.9
probability of mutation	:	0.0
crowding factor	:	3
generation gap	:	0.1

The parameters are held constant across all runs. To minimize the stochastic error due to the selection procedure, stochastic remainder selection method is used, and to judge the appropriate diversity in the population with the schemes alone, the probability of mutation is set to zero. In the crowding scheme, the parameters (CF and G) are set to the values used by De Jong (1975). In the sharing schemes, a triangular sharing distribution ($\alpha = 1$) is used. The sharing parameter σ_{share} used in the simulation is calculated using the derived relations. For the functions F1 and F2, $\mathrm{x}_{max} = 1$, $\mathrm{x}_{min} = 0$, $q = 5$, and $\ell = 30$, so that phenotypic $\sigma_{\text{share}} = \frac{1-0}{2\times5} = 0.1$ and genotypic σ_{share} is calculated as $\sum_{i=1}^{\sigma_{\text{share}}} \binom{30}{i} = \frac{1}{5}2^{30}$; or, $13 < \sigma_{\text{share}} < 14$. Function F3 has four peaks in the rectangle $-6.0 \leq x_1, x_2 \leq 6.0$. Phenotypic and genotypic sharing parameter are calculated using Equations 3 and 5 and are found to be 4.24 and 13 respectively. To calculate the performance measures, five runs are carried out with different initial populations generated at random, and an average statistic is calculated for each of the three performance measures. To make a fair comparison, however, the same five initial populations are used for each scheme.

Figure 1 shows the distribution of 100 individuals plotted on the function F1 after 200 generations with crowding, phenotypic sharing, and genotypic sharing. A visual comparison on the plots reveals that crowding is unable to maintain stable subpopulations at all the peaks of the function, whereas the sharing methods are able to cluster trials around all five peaks of the function.

The on-line performance measure on the function F1 after 200 generations with crowding, phenotypic sharing, and genotypic sharing is found to be 0.892, 0.863, and 0.866 respectively. Though, all three schemes show a high value of on-line performance measure on function F1, recall that in the crowding scheme, the individuals converge to only two of the five peaks (Figure 1), whereas both the sharing methods are able to converge to all five peaks, and still attain high on-line performance. All three schemes achieve similarly high off-line performance values indicating good convergence to one or more peaks on F1 although the corresponding plot is not reproduced here.

To determine how well the schemes have distributed the individuals to the different peaks, the deviation measure described in the previous section is calculated and is shown in Figure 2. Since all peaks are of the same size, the expected number (μ_i) of trials in each peak is $\frac{100}{5} = 20$, and the variance $\sigma_i^2 = 100 \times 0.2 \times (1 - 0.2) = 16$. The expected number and variance corresponding to the individuals not representing any peak are calculated using Equation 7 and 8:

$$\mu_6 = 0, \quad \text{and} \quad \sigma_6^2 = \sum_{i=1}^{5} 100 \times 0.2 \times (1 - 0.2) = 80.$$

The low deviation measure values with sharing methods indicate a near-ideal distribution of trials over the peaks on F1. It can be inferred from the plots that sharing methods are better able to maintain and converge to stable subpopulations at the peaks of function F1.

Simulations on function F2 also produces interesting results. Figure 3 shows representative populations of points at generation 200 generated by all the schemes. Crowding cannot maintain a spread of trials on all the peaks and converges to one peak. Phenotypic sharing is able to maintain subpopulations proportionate to their peak fitness values at all five peaks, whereas genotypic sharing is unable to cluster trials at the smaller peaks. It is found that when dealing with functions having unequal peaks, genotypic sharing sometimes cannot maintain subpopulations at the peaks of lower fitness values. There is a critical ratio of the peak fitness values above which genotypic sharing is unable to maintain subpopulations at lower peaks. A simplified analysis of this ratio is presented in a later section.

Since F2 has unequal peaks, a high value of on-line performance measure does not imply good convergence at all the peaks, and the corresponding plot is not presented here. A better understanding of the performance of the schemes may be obtained from the deviation measure. The expected number of individuals, μ_i, and the variance, σ_i^2, of the individuals on each peak are tabulated in Table 1. Using these values in Equation 9, the chi-square-like deviation measure may be calculated at any generation and is shown in Figure 4. Phenotypic sharing performs well on F2 as demonstrated by its low deviation value.

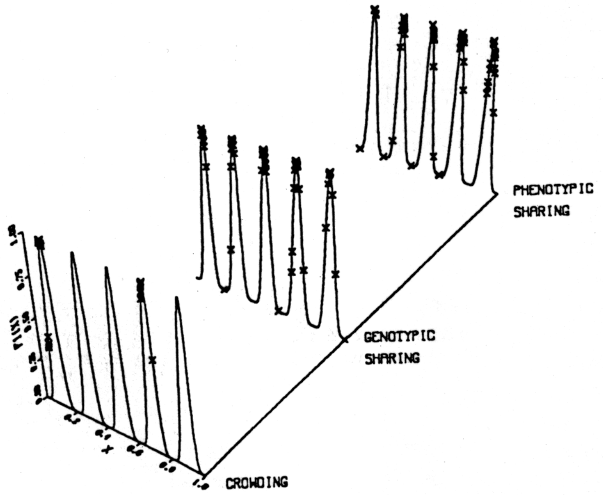

Figure 1. State of the population on function F1 after 200 generations. GAs with sharing distribute trials to all five peaks; a GA with crowding converges to two peaks.

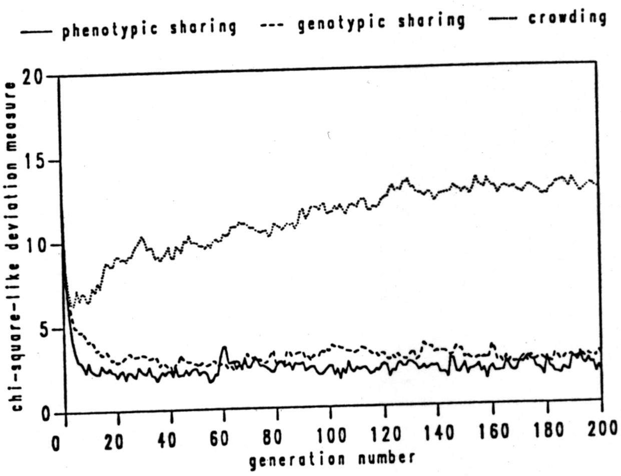

Figure 2. Comparison of the chi-square-like deviation measure of all three schemes on function F1. GAs with sharing show a small deviation from the ideal distribution.

Table 1. Expected number and variance of different subspaces on F2.

subspace	f_i	μ_i	σ_i^2
1st peak	1.000	30	21.00
2nd peak	0.917	27	19.95
3rd peak	0.707	21	16.72
4th peak	0.458	14	11.86
5th peak	0.250	8	6.94
non-peak		0	76.47

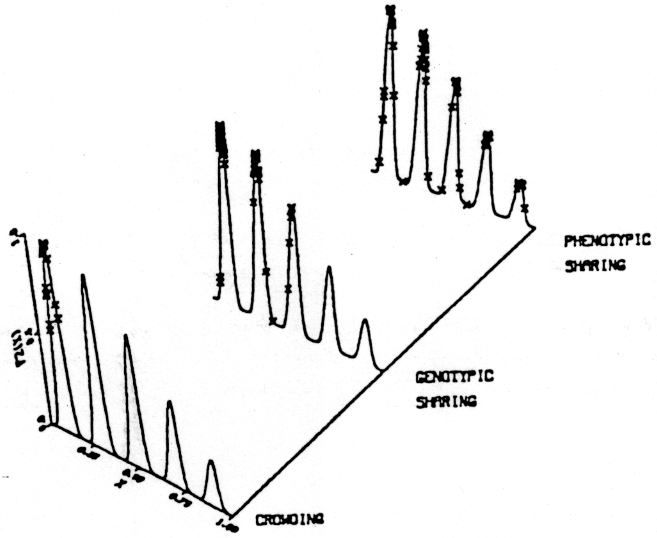

Figure 3. State of the population on function F2 after 200 generations. A GA with phenotypic sharing distributes trials to all five peaks, a GA with genotypic sharing distributes trials to three peaks, and a GA with crowding distributes trials to a single peak.

Function F3 is a two-parameter function and has four peaks in the solution space. The sketches showing the distribution of trials on the contours of the function are presented in the original study (Deb, 1989). The sketches show that crowding is able to converge to two peaks, whereas sharing is able to converge to all four peaks. The on-line performance value on the function F3 after 200 generations with crowding, phenotypic sharing, and genotypic sharing is found to be 0.851, 0.884, and 0.811 respectively. All the schemes show good convergence to one or more peaks of the function. Figure 5 shows chi-square-like measure on F3 for all these methods. Small deviations for both sharing methods confirms the successful formation of niches in the population. The difference between phenotypic and genotypic sharing is significant, however, with phenotypic sharing coming closer to ideal distribution.

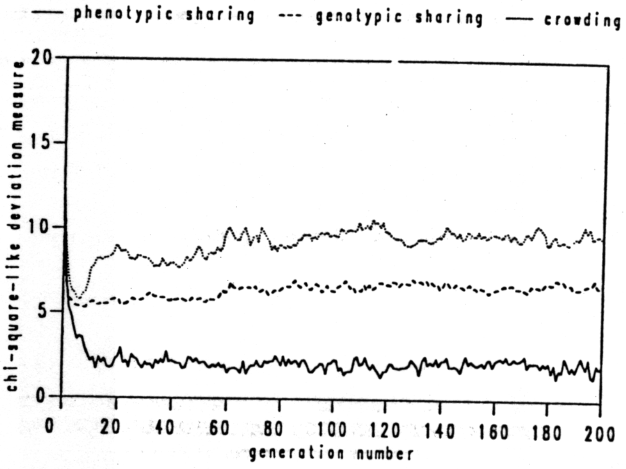

Figure 4. Comparison of the chi-square-like deviation measure of all three schemes on function F2. A GA with phenotypic sharing shows a small deviation from the ideal distribution.

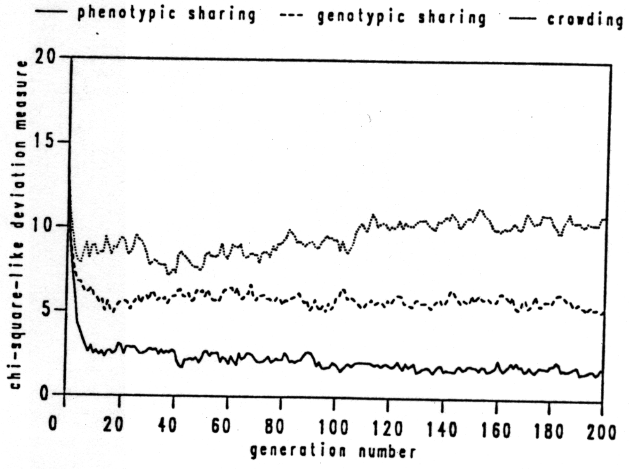

Figure 5. Comparison of the chi-square-like deviation measure of the schemes on function F3. GAs with sharing show a smaller deviation from the ideal distribution than does crowding.

5 A Limitation of Genotypic Sharing

On functions with peaks of unequal value, genotypic sharing is sometimes unable to maintain stable subpopulations at the peaks of relatively low value. This phenomenon can be explained by considering a simple function having two competing peaks with payoff values f_1 and f_2 with $f_1 \geq f_2$ corresponding to the strings s_1 and s_2 as idealized in Figure 6. In genotypic sharing, the metric d_{ij} is defined as the Hamming distance between the strings. Considering a triangular sharing function with $\alpha = 1$, the genotypic sharing function Sh_{ij} between any two individuals i and j in the population can be written as

$$Sh_{ij} = \begin{cases} 1 - \left(\dfrac{d_{ij}}{\sigma_{\text{share}}}\right), & \text{if } d_{ij} < \sigma_{\text{share}}; \\ 0, & d_{ij} \geq \sigma_{\text{share}}. \end{cases} \quad (10)$$

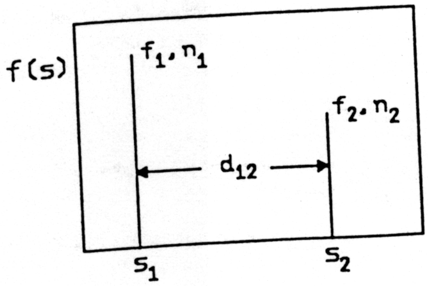

Figure 6. Idealized bimodal function with peaks corresponding to strings s_1 and s_2.

Assume that after a substantial number of generations, all the individuals have settled in one of the two peaks with n_1 and n_2 individuals in peaks 1 and 2 respectively such that the total population size $n = n_1 + n_2$. The niche count of an individual corresponding to each niche is $m'_1 = n_1 + n_2 Sh_{12}$ and $m'_2 = n_2 + n_1 Sh_{12}$. Therefore, the shared fitness values are $f'_1 = \frac{f_1}{n_1 + n_2 Sh_{12}}$ and $f'_2 = \frac{f_2}{n_2 + n_1 Sh_{12}}$ respectively. To establish stable subpopulations on both the peaks, the shared fitness values of an individual in either peak must be equal; otherwise, only the one with higher shared fitness will be selected. Therefore, setting $f'_1 = f'_2$,

$$\frac{f_1}{f_2} = \frac{n_1 + n_2 Sh_{12}}{n_2 + n_1 Sh_{12}} = \gamma. \quad (11)$$

The parameter γ is the peak fitness ratio and as $f_1 \geq f_2$, the parameter γ has a value greater than or equal to one. Rearranging, the above equation then takes the form

$$\frac{n_2}{n} = \frac{1 - \gamma Sh_{12}}{(1+\gamma)(1 - Sh_{12})}. \quad (12)$$

It is clear from the above equation that individuals exist in the second peak (with lower fitness value) only if the numerator of the above equation is positive:

$$1 - \gamma Sh_{12} \geq 0,$$

$$Sh_{12} \leq \frac{1}{\gamma}, \quad \text{or,}$$

$$d_{12} \geq (1 - \frac{1}{\gamma})\sigma_{\text{share}}. \quad (13)$$

Therefore, there is a lower bound on the Hamming difference between the strings representing the peaks below which genotypic sharing is incapable of forming any subpopulation on the smaller peak. This bound is dependent on the fitness-ratio of the peaks, γ, and the parameter σ_{share}. On the other hand, for a particular peak fitness ratio, the positions of the peaks in the gene space should be such that the Hamming difference between them is more than that derived from the right side of Equation 13. It is interesting to note that, for functions having peaks of equal fitness values, $\gamma = 1$, and using this value in Equation 13, $d_{12} \geq 0$. That is, with peaks of equal value, any bit difference between the strings representing the peaks is sufficient to form stable subpopulations around the peaks. Therefore, in dealing with functions having peaks of equal fitness value, genotypic sharing allocates trials successfully over all the peaks.

In the next section, the performance of sharing schemes is further improved by imposing the concept of mating restriction.

6 Adding a Mating Restriction Scheme

Once the sharing scheme clusters subpopulations of trials at the peaks, crossover between strings on different peaks may produce new strings that do not represent any peak. The presence of these lethal strings in the population degrades the on-line performance of the process. Therefore, to achieve better performance on multimodal functions, crossover between strings of different peaks must be reduced. In nature, this is achieved by creating separate species (or subpopulation) corresponding to each niche (or peak) in the solution space and restricting the mating between species. A number of methods have been suggested and developed to induce species into the population using some sort of restriction in mating (Booker, 1982; Deb, 1989; Goldberg, 1989). This paper considers a simple mating restriction scheme based on the phenotypic distance between the mating individuals. In the previous section, it has been shown that a population can be divided into as many subpopulations as there are peaks by suitably selecting a parameter σ_{share}. A similar parameter can also be defined to create subpopulations that constitute a species. When the distance-metric considered in both these cases are measured in the decoded parameter space, so-called *phenotypic mating restriction*, the individuals with the corresponding distance-metric less than σ_{share} are shared and the individuals with distance-metric less than σ_{mating} are mated. To keep the analysis simple, the parameters σ_{share} and σ_{mating} are set equal. The mating restriction method imposed on the individuals is as follows:

> To find a mating companion of an individual, if an individual within a distance of σ_{mating} is found, mating is performed, otherwise another individual is tried. If no such individual is found in the population, a random individual is chosen.

This scheme has been implemented and applied to the test functions defined earlier. This paper presents the simulation results on Function F1 only; more detail results are available in the original work (Deb, 1989). Figure 7 shows the reduction in lethal individuals due to the mating restriction. Figure 8 shows the on-line performance measure of phenotypic sharing alone and phenotypic sharing with phenotypic mating restriction. The average value of the chi-square-like measure on function F1 after 200 generations with phenotypic sharing alone and phenotypic sharing with phenotypic mating restriction is found to be 2.169 and 0.487 respectively.

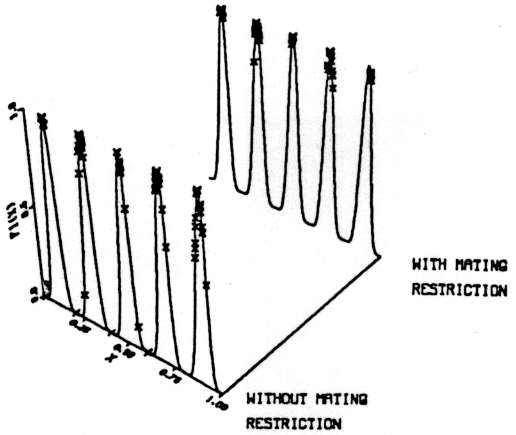

Figure 7. Distribution of trials on function F1 with phenotypic sharing alone and phenotypic sharing with phenotypic mating restriction. Sharing with mating restriction reduces the number of lethal trials.

An increase in the value of the on-line performance measure and simultaneously, a decrease in the value the chi-square-like deviation measure reveal that sharing with mating restriction improves the on-line performance of sharing and also better distribute trials at peaks in proportion to peak fitness values.

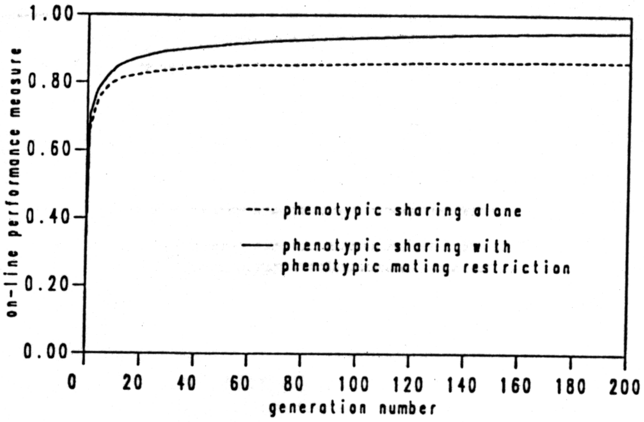

Figure 8. Comparison of on-line performance measure on F1 with phenotypic sharing alone and phenotypic sharing with phenotypic mating restriction.

Tag-template mating restriction has also been considered in the original study (Deb, 1989), but this work is beyond the scope of this paper.

7 Conclusions

In artificial genetic search, multimodal functions are optimized by creating separate niches in a problem, thereby forcing individuals within the same niche to fight for limited population slots. This paper has compared the crowding method of De Jong and the sharing function method of Goldberg and Richardson by analyzing their performance on three multimodal functions. Simulation runs have shown that the sharing is better able to allocate individuals to the peaks. Of the two sharing function methods, phenotypic sharing and genotypic sharing, it has been observed that genotypic sharing is sometimes unable to maintain subpopulations at sub-optimal peaks. An analysis has shown that a minimum Hamming distance is required to separate peaks of differing fitness ratio.

In multimodal function optimization, crossover between individuals of different peaks may produce individuals that do not belong to any peak. The presence of these individuals in the population degrades the on-line performance of a scheme. Therefore, to further improve the performance of niche methods, a mating restriction scheme has been used. The experimental results with a simple mating restriction scheme have demonstrated improvement in the on-line performance of phenotypic sharing.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant MSM-8451610.

References

Booker, L. B. (1982). Intelligent behavior as an adaptation to the task environment. (Doctoral Dissertation, Technical Report No. 243. Ann Arbor: University of Michigan, Logic of Computers Group). *Dissertation Abstracts International, 43*(2), 469B. (University Microfilms No. 8214966)

De Jong, K. A. (1975). An analysis of the behavior of a class of genetic adaptive systems. (Doctoral dissertation, University of Michigan). *Dissertation Abstracts International, 36*(10), 5140B. (University Microfilms No. 76-9381)

Deb, K. (1989). *Genetic algorithms in multimodal function optimization,* Master's thesis, TCGA Report No. 88002, University of Alabama, Tuscaloosa.

Goldberg, D. E. (1983). Computer-aided gas pipeline operation using genetic algorithms and rule learning. (Doctoral dissertation, University of Michigan). *Dissertation Abstracts International, 44*(10), 3174B. (University Microfilms No. 8402282)

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning.* Reading, MA: Addison-Wesley.

Goldberg, D. E., and Richardson, J. (1987). Genetic algorithms with sharing for multimodal function optimization. *Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms,* 41-49.

Goldberg, D. E., and Segrest, P. (1987). Finite Markov chain analysis of genetic algorithms. *Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms,* 1-8.

Goldberg, D. E., and Smith, R. E. (1987). Nonstationary function optimization using genetic algorithms with dominance and diploidy. *Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms,* 59-68.

Holland, J. H. (1975). *Adaptation in natural and artificial systems.* Ann Arbor: The University of Michigan Press.

Reklaitis, G. V., Ravindran, A., and Ragsdell, K. M. (1983). *Engineering optimization methods and applications.* New York: Wiley.