

Migration Policies, Selection Pressure, and Parallel Evolutionary Algorithms

Erick Cantú-Paz

*Department of Computer Science and Illinois Genetic Algorithms Laboratory,
University of Illinois at Urbana-Champaign, Urbana, IL
cantupaz@illigal.ge.uiuc.edu**

Abstract. This paper investigates how the policy used to select migrants and the individuals they replace affects the selection pressure in parallel evolutionary algorithms (EAs) with multiple populations. The four possible combinations of random and fitness-based emigration and replacement of existing individuals are considered. The investigation follows two approaches. The first is to calculate the takeover time under the four migration policies. This approach makes several simplifying assumptions, but the qualitative conclusions that are derived from the calculations are confirmed by the second approach. The second approach consists on quantifying the increase in the selection intensity. The selection intensity is a domain-independent adimensional quantity that can be used to compare the selection pressure of common selection methods with the pressure caused by migration. The results may help to avoid excessively high (or low) selection pressures that may cause the search to fail, and offer a plausible explanation to the frequent claims of superlinear speedups in parallel EAs.

Keywords: Multiple populations, multiple demes, island model, migration rate, emigrants, immigrants

* Current address: Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA cantupaz@llnl.gov



1. Introduction

Parallel evolutionary algorithms (EAs) have received considerable attention because of their potential to reduce the execution time in complex applications. One common method to parallelize EAs is to use multiple demes (populations) that occasionally exchange some individuals in a process called migration. A specification of a parallel EA with multiple demes defines the size and number of the demes, the topology of the connections between them, the migration rate (the fraction of the population that migrates), the frequency of migrations, and the policy to select emigrants and to replace existing individuals with incoming migrants. The importance of these parameters on the quality of the search and on the efficiency of the algorithms has been recognized for a long time (Grefenstette, 1981; Grosso, 1985; Tanese, 1987).

This paper shows that the migration policy affects considerably the selection pressure, and that it influences significantly the speed of convergence of the algorithm. Although it has been recognized before that fitness-based selection of emigrants and replacements may increase the selection pressure (Whitley, 1993; Whitley et al., 1999; Cantú-Paz, 1999), its impact on the algorithm's convergence is not well understood. The objective of this paper is to quantify accurately the additional selection pressure caused by migration. This leads to accurate predictions of the number of generations until convergence. In addition, understanding the effect of the migration policies on the selection pressure is important because excessively slow or fast convergence rates may cause the search to fail (Goldberg et al., 1993; Thierens and Goldberg, 1993). If selection is too weak the population may drift aimlessly for a long time, and the quality of the solutions found is not likely to be good. On the other hand, rapid convergence is desirable, but an excessively fast convergence may cause the EA to converge prematurely to a suboptimal solution.

This investigation also offers one of the first theoretical explanations for the frequent claims of superlinear speedups in parallel EAs (see for example (Tanese, 1989; Belding, 1995; Punch, 1998)). These controversial claims imply that the parallel EA requires less total effort to reach the same solution than a serial EA (Punch, 1998). However, we are left with a perplexing question: What caused the reduction in the required work? The answer offered by this paper is that the number of generations until convergence is reduced by the additional selection pressure caused by some migration policies.

There are two alternatives to select the individuals that emigrate from a deme. They can be chosen randomly or by selecting the best individuals in a deme. Likewise, there are two choices to replace existing

individuals in the receiving deme with the incoming migrants: choose randomly or replace the worst. Each of the four combinations of migrant selection and replacement causes a different selection pressure, and this paper examines them from two perspectives. First, we compute the takeover time, and then we calculate the increase in the selection intensity. The calculations in this paper assume that migration occurs every generation, which is an upper bound on the migration frequency. Less frequent migrations are expected to have a lower impact on the convergence of the algorithm. In addition, the calculations assume that the migrants are copies of the individuals selected to migrate. This differs from natural populations, but is the most frequently-used form of migration in EAs.

The paper is organized as follows. The next section reviews common selection methods and explains intuitively the concept of selection pressure. Section 3 summarizes previous work on takeover times for simple GAs and extends the calculations to consider the four migration policies. The calculation of the selection intensity caused by the different migration policies is in section 4, and the results of experiments that validate the accuracy of the models are in section 5. Section 6 has a brief discussion of superlinear speedups and illustrates how some of them can be explained by an increase in selection pressure. Finally, section 7 summarizes the findings of this study and discusses open issues for future research.

2. Selection Methods and Selection Pressure

There are many different mechanisms used in evolutionary algorithms to select the parents of the next generation. They can be classified roughly into two groups: fitness-proportionate and rank-based selection.

Fitness-proportionate methods select individuals probabilistically depending on the ratio of their fitness and the average fitness of the population. These methods are amongst the earliest methods used in EAs. Some examples are roulette-wheel selection (or also called proportionate selection) (Holland, 1975), stochastic remainder selection (Booker, 1982), and stochastic universal selection (Baker, 1987). Roulette-wheel selection uses a simulated biased roulette wheel with slots that are sized according to the fitness of each individual. The roulette is spinned once for each individual to be selected. The other schemes were proposed to reduce the stochastic error associated with spinning the roulette wheel numerous times.

Some common rank-based selection methods are linear ranking (Baker, 1985), tournament selection (Brindle, 1981), $(\mu^+ \lambda)$ selection (Schwefel, 1981), and truncation selection (Mühlenbein and Schlierkamp-Voosen, 1993). In linear ranking selection, individuals are selected with a probability that is linearly proportional to the rank of the individuals in the population. The desired expected number of copies of the best (n^+) and worst ($n^- = 2 - n^+$) individuals are supplied as parameters to the algorithm. In tournament selection, a random sample of s individuals is selected (with or without replacement), and the best individual in the sample is selected. The process is repeated until the mating pool is filled. In $(\mu + \lambda)$ selection, λ offspring are created from μ parents, and the μ best individuals out of the union of parents and offspring are selected. In (μ, λ) selection ($\lambda \geq \mu$) the μ best offspring are selected to survive. Truncation selection selects the top $1/\tau$ of the population and creates τ copies of each individual. It is equivalent to (μ, λ) selection with $\mu = \lambda/\tau$.

All of these selection mechanisms have the same purpose of creating more copies of the individuals with higher fitness than of those with low fitness. However, the selection mechanisms differ in the manner in which they allocate copies to the fittest individuals. A selection method has a higher selection pressure than another if it makes more copies of the best individuals, thereby eliminating rapidly low-fit individuals. A strong selection method reaches equilibrium faster than a weaker method, but it also sacrifices genetic diversity that may be needed to find an adequate solution.

The parameters of selection methods regulate the selection pressure, which in turn determines how fast the algorithms converge. We shall see that the parameters of migration also affect the selection pressure. The speed of convergence of different selection schemes was first studied by Goldberg and Deb (1991), who introduced the concept of takeover time. The takeover time is the number of generations that selection alone requires to replicate a single individual of the best class until the population is full. The next section of this paper extends Goldberg and Deb's analysis to consider different migration policies. Later, Mühlenbein and Schlierkamp-Voosen (1993) used concepts from population genetics to study convergence properties of a particular EA and introduced the use of the selection intensity to study the convergence of selection schemes. Section 4 calculates the additional selection intensity that originates from migration. Of course, many others have studied and compared different selection methods used in EAs (for example, see the papers by Bäck (1994) and by Hancock (1997))

3. Takeover Times

This section is based on Goldberg and Deb's (1991) analysis of the takeover times of tournament selection, but similar calculations may be performed for other selection schemes. This section considers a simplified population model with only two classes of individuals: good and bad. We may think of the good individuals as representatives of the global solution, while the bad individuals are any other lesser solution. The calculations of takeover time only consider the effect of selection on the growth of good solutions and ignore other operators such as recombination and mutation.

Let P_t denote the proportion of good individuals in the population at time t , and $Q_t = 1 - P_t$ denote the proportion of bad individuals. In the particular case of tournament selection of size s , a bad individual will survive only if all the participants in the tournament are bad:

$$Q_{t+1} = Q_t^s. \quad (1)$$

Substituting $P = 1 - Q$ gives the proportion of good individuals:

$$P_{t+1} = 1 - (1 - P_t)^s. \quad (2)$$

The extensions below assume that migration occurs every generation, and that it occurs after selection, so its effect may be accounted for by adding a policy-dependent term to equation 2.

The simplified analysis of this section does not consider the origin of good migrants. For example, the analysis does not distinguish if a deme receives ten good migrants from one neighbor or five good migrants from two neighbors. We recognize that the topology is one of the most important parameters of migration, because it affects considerably both the solution quality and the cost of communications (Cantú-Paz and Goldberg, 1999). However, the takeover time calculations are not concerned with obtaining a good solution or with minimizing the communications, but rather with investigating how fast a good solution dominates a population once it is found. Subsequent sections follow another approach that makes fewer simplifying assumptions about the population model and the topology of communications. However, the qualitative observations made possible by the simple takeover time calculations will remain valid.

Out of the four migration policies, the easiest case to model is when good migrants replace bad individuals. With this policy, on every generation the proportion of good individuals increases by the migration rate ρ (which is the fraction of the population that migrates). Therefore, a

difference equation for this case can be obtained easily, just by adding ρ to Goldberg and Deb's equation (eq. 2):

$$P_{t+1} = 1 - (1 - P_t)^s + \rho. \quad (3)$$

In the case where good individuals migrate and the replacements are chosen randomly, the migrants may replace good or bad individuals. If good migrants replace good individuals, then the proportion of good individuals in the receiving deme remains unchanged. Thus, we are interested in calculating how many bad individuals are replaced. The probability of replacing bad individuals is equal to their proportion in the population after selection, $Q_{t+1} = Q_t^s = (1 - P_t)^s$, and therefore the proportion of bad individuals that is replaced by the good migrants is $\rho(1 - P_t)^s$. Adding this to equation 2 we obtain

$$P_{t+1} = 1 - (1 - P_t)^s + \rho(1 - P_t)^s. \quad (4)$$

We can use a similar idea to examine the case when random migrants replace bad individuals. The proportion of good individuals in the receiving deme will increase by the good migrants, so we are interested in calculating how many migrants are good. Since migrants are chosen uniformly at random, the proportion of good migrants is the same as the proportion of good individuals present in a deme, $1 - (1 - P_t)^s$. Therefore, the proportion of good individuals at the receiving deme is incremented by the good migrants as follows:

$$\begin{aligned} P_{t+1} &= 1 - (1 - P_t)^s + \rho(1 - (1 - P_t)^s) \\ &= (1 + \rho)(1 - (1 - P_t)^s). \end{aligned} \quad (5)$$

When random migrants replace random individuals the proportion of good individuals in each deme is expected to be the same over time, choosing migrants and replacements randomly does not have any effect on the takeover times.

To find the takeover times of the different migration policies we simply iterate the difference equations and count the number of iterations until P_t reaches or exceeds 1. The starting point for the equations is $P_0 = 1/n$ (there is a single good individual in the population). Figure 1 shows the takeover times in demes with $n = 10000$ individuals and pairwise tournament selection ($s = 2$). The plots illustrate how as the migration rate increases the convergence is faster, and that the fastest convergence occurs when good migrants replace bad individuals, which is a frequently used migration policy (e.g., (Grefenstette, 1981; Tanese,

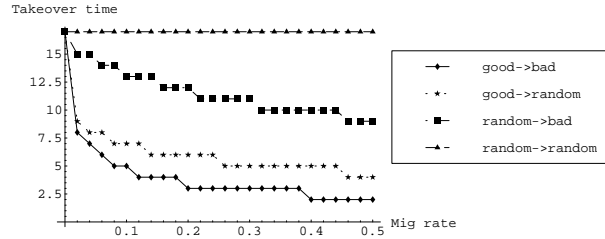


Figure 1. Takeover times using different migration policies and varying the migration rate.

1987; Mühlenbein, 1991; Lin et al., 1997)). The slowest convergence occurs when both migrants and replacements are chosen randomly. A GA with a single population would converge in exactly the same time as this case.

These simple calculations suggest that the difference between the fastest and slowest convergence times is quite large, and therefore the migration policy must be taken into account when designing parallel GAs. The analysis also suggests that the choice of migrants is a greater factor in the convergence speed than the choice of replacements. A similar behavior is observed in the next sections.

4. Selection Intensity

This section follows a different approach to estimate the convergence times of parallel GAs. The assumptions used here are more realistic than in the previous section, and the results are very accurate predictions of the convergence time. First, the concept of selection intensity is defined and closed-form expressions for the selection intensity of common selection methods is provided as a reference. Then, the section presents the derivation of the additional selection intensity caused by migration. We show how these values can be used to predict the convergence times, and the next section verifies empirically the accuracy of the predictions.

The first observation needed to calculate the intensity of a selection method is that the average fitness of the individuals that are selected to survive is greater than the average fitness of the population. The magnitude of this increase depends on the selection method being used, and it can be quantified as the *selection differential*

$$s^t = \bar{f}_s^t - \bar{f}^t, \quad (6)$$

Table I. Selection intensity for common selection schemes.

Selection Method	Parameters	I
Tournament	s	$\mu_{s:s}$
(μ, λ)	μ, λ	$\frac{1}{\mu} \sum_{i=\lambda-\mu+1}^{\lambda} \mu_{i:\lambda}$
Linear Ranking	n^+	$(n^+ - 1) \frac{1}{\sqrt{\pi}}$
Proportional	σ_t, μ_t	σ_t / μ_t

which is the difference between the mean fitness of the selected individuals and the mean fitness of the population (Mühlenbein and Schlierkamp-Voosen, 1993). The superscript t denotes the generation number. Assuming that the fitness of the population has a normal distribution, the selection differential can be calculated as

$$s^t = I\sigma_t, \quad (7)$$

where σ_t is the standard deviation of the population at time t , and the factor I is the *selection intensity*. The selection intensity of some common selection schemes has been calculated analytically. Bäck (1995) and Miller and Goldberg (1995) independently derived the selection intensity for tournament selection, and Bäck (1995) also derived I for (μ, λ) selection. Blickle and Thiele (1996) calculated the intensity of linear ranking, and Mühlenbein and Schlierkamp-Voosen (1993) calculated I for proportional selection. Table I contains the known selection intensities (adapted from (Miller and Goldberg, 1996)). Note that I is independent of the distribution of the current population, except for proportional selection.

Calculating the standard deviation is more complicated, and it depends on the fitness function. For the case of a l -bit OneMax function $f = \sum_{i=0}^l x_i$ (x_i is the i -th bit of the string x), uniform crossover (Syswerda, 1989) creates an approximately binomial fitness distribution with probability P_t , where P_t is the proportion of bits set to one in generation t . Therefore, the standard deviation may be calculated as (Mühlenbein and Schlierkamp-Voosen, 1993; Thierens and Goldberg, 1994):

$$\sigma_t = \sqrt{l P_t(1 - P_t)}. \quad (8)$$

The selection differential $\bar{f}^{t+1} - \bar{f}^t = \sigma_t \cdot I$ can be written as

$$P_{t+1} - P_t = \frac{I}{\sqrt{l}} \sqrt{P_t(1 - P_t)}.$$

This can be approximated as a differential equation that can be solved to obtain the proportion of correct bits at time t as

$$P_t = 0.5 \left(1 + \sin \left(\frac{I}{\sqrt{l}} t + \arcsin(2P_0 - 1) \right) \right), \quad (9)$$

where P_0 is the initial proportion of bits correct. In the case of the OneMax function this is usually 0.5. The number of generations until convergence can be calculated making $P_t = 1$ and solving for t to obtain

$$G = \left(\frac{\pi}{2} - \arcsin(2P_0 - 1) \right) \frac{\sqrt{l}}{I}, \quad (10)$$

which in the usual case when $P_0 = 0.5$ simplifies to $G = \frac{\pi}{2} \frac{\sqrt{l}}{I}$. We shall use the two equations above to verify the calculations of the selection intensity caused by migration.

4.1. MIGRATION AND SELECTION INTENSITY

The remainder of the section follows closely the notation and method used by Bäck (1995) in his study of (μ, λ) selection. This should not imply that fitness-based migration is equivalent to (μ, λ) selection. In fact it is not. Bäck's paper provides a good framework and some useful approximations that we adapt here for our own purposes.

To calculate the selection intensity, we must calculate the selection differential between the mean fitness of a deme after migration $\bar{f}^{t'}$ and before migration \bar{f}^t :

$$s^t = \bar{f}^{t'} - \bar{f}^t. \quad (11)$$

The average fitness before migration is simply $\bar{f}^t = \frac{1}{n} \sum_{i=1}^n f_i^t$, where f_i^t is the fitness of the i -th individual of the population at generation t . The average fitness after migration can be written as a weighted sum of the average fitness of the migrants and the average fitness of the individuals that survive migration (i.e., are not replaced by migrants). Let δ denote the degree of the topology (the number of neighbors of a deme), and $m = \rho n$ the number of migrants from *one* deme. We can write

$$\bar{f}^{t'} = \frac{1}{n} (\delta m \bar{f}_{mig}^t + (n - \delta m) \bar{f}_{sur}^t), \quad (12)$$

where \bar{f}_{mig}^t is the mean fitness of migrants from one deme, and \bar{f}_{sur}^t is the mean fitness of the $(n - \delta m)$ survivors. Similarly we can write

$$\bar{f}^t = \frac{1}{n} (\delta m \bar{f}^t + (n - \delta m) \bar{f}^t). \quad (13)$$

Grouping similar terms we can decompose the selection differential into two parts. One that corresponds to the selection of emigrants s_e^t and another that corresponds to the selection of replacements s_r^t :

$$\begin{aligned} s^t &= s_e^t + s_r^t \\ &= \frac{1}{n} \delta m (\bar{f}_{mig}^t - \bar{f}^t) + \frac{1}{n} (n - \delta m) (\bar{f}_{sur}^t - \bar{f}^t). \end{aligned} \quad (14)$$

By writing the selection differential in this way, we can separate the calculation of the selection intensity into two independent steps. First, we consider the intensity caused by selecting the best individuals to emigrate. We shall see that selecting emigrants randomly has no effect on the selection intensity. Later, similar calculations are performed to calculate the intensity caused by replacing individuals.

The major assumption that we make is that the fitness values of the population at time t have a normal distribution. Under this assumption, the fitness values f_i^t can be interpreted as samples of random variables F_i^t with a common distribution $N(\bar{f}^t, \sigma_t)$.

We may arrange the random variables in increasing order as

$$F_{1:n}^t \leq F_{2:n}^t \leq \dots \leq F_{n:n}^t.$$

These are the order statistics of the F_i^t variables, and we can use them to calculate the average fitness of the emigrants and the survivors. Without loss of generality we assume a maximization problem. The mean fitness of the $m = \rho n$ best individuals selected to migrate from *one* deme is

$$\bar{f}_{mig}^t = \frac{1}{m} \sum_{i=n-m+1}^n E(F_{i:n}^t). \quad (15)$$

The random variables can be normalized as

$$Z_{i:n} = \frac{F_{i:n}^t - \bar{f}^t}{\sigma_t} \sim N(0, 1),$$

and the average fitness of the migrants may be rewritten in terms of the normalized variables

$$\begin{aligned}
\bar{f}_{mig}^t &= \frac{1}{m} \sum_{i=n-m+1}^n (E(Z_{i:n})\sigma_t + \bar{f}^t) \\
&= \sigma_t \cdot \frac{1}{m} \sum_{i=n-m+1}^n E(Z_{i:n}) + \bar{f}^t.
\end{aligned} \tag{16}$$

Now, we can calculate the selection differential caused by the migrants as

$$s_e^t = \frac{\delta m}{n} (\bar{f}_{mig}^t - \bar{f}^t) = \frac{1}{n} \cdot \sigma_t \cdot \delta \sum_{i=n-m+1}^n E(Z_{i:n}). \tag{17}$$

Since the selection differential is also defined as $s^t = I \cdot \sigma_t$ (equation 7), the selection intensity caused by selecting the best individuals to emigrate is

$$I_e = \frac{1}{n} \cdot \delta \cdot \sum_{i=n-m+1}^n E(Z_{i:n}). \tag{18}$$

The expected value of the i -th order statistic of a sample of size n is defined as

$$\mu_{i:n} = E(Z_{i:n}) = n \binom{n-1}{i-1} \int_{-\infty}^{\infty} z \phi(z) \Phi^{i-1}(z) [1 - \Phi(z)]^{n-i} dz, \tag{19}$$

where $\phi(z) = \exp(-z^2/2)/\sqrt{2\pi}$ and $\Phi(z) = \int_{-\infty}^z \phi(x) dx$ are the PDF and CDF respectively of a standard normal distribution with mean 0 and standard deviation of 1. The values of $\mu_{i:n}$ are computationally expensive to calculate, but they are tabulated for $n \leq 400$ (Harter, 1970). Nevertheless, computing the sum in equation 18 can be cumbersome, but fortunately the following approximation exists¹ (Burrows, 1972; Bäck, 1995):

$$\sum_{i=n-m+1}^n \mu_{i:n} \approx n \phi(\Phi^{-1}(1 - \rho)), \tag{20}$$

and therefore equation 18 can be approximated as

¹ Bäck shows that for $n > 50$ the approximation is indistinguishable from the real values.

$$I_e \approx \delta\phi(\Phi^{-1}(1 - \rho)). \quad (21)$$

It is important to realize that the selection intensity is an adimensional quantity that does not depend on the fitness function or on the generation t . The only assumption made to calculate the intensity is that the fitness values have a normal distribution, but any other distribution may be used as long as $E(F_{i:n})$ may be computed (by substituting the appropriate PDF and CDF in equation 19). The selection intensity depends on the population size (see equation 18), but its approximation depends only on the migration rate.

In the case where individuals are chosen randomly to emigrate, the expected fitness of the migrants is the same as the mean fitness of the population, and therefore the selection differential is $s^t = \bar{f}_{mig}^t - \bar{f}^t = 0$, and there is no additional selection intensity ($I_e = 0$). However, there may be an increase in the overall selection intensity if the migrants replace the worst individuals in the target deme. The replacement of individuals is treated in the following paragraphs.

Replacing the *worst* individuals in a deme with migrants causes an increase in the average fitness of the deme. In a manner similar as above, we can calculate the mean fitness of the individuals that survive (i.e., are not replaced by the δm migrants) as

$$\begin{aligned} \bar{f}_{sur}^t &= \frac{1}{n - \delta m} \sum_{i=\delta m+1}^n E(F_{i:n}) \\ &= \sigma_t \cdot \frac{1}{n - \delta m} \sum_{i=\delta m+1}^n \mu_{i:n} + \bar{f}^t. \end{aligned} \quad (22)$$

In this case the response to selection is the difference between the mean of the individuals that survive and the mean fitness of the population:

$$\begin{aligned} s_r^t &= \frac{1}{n} (n - \delta m) (\bar{f}_{sur}^t - \bar{f}^t) \\ &= \sigma_t \cdot \frac{1}{n} \sum_{i=\delta m+1}^n \mu_{i:n} \\ &= \sigma_t \cdot I. \end{aligned} \quad (23)$$

Therefore, the selection intensity caused by choosing the worst individuals to be replaced is

$$I_r = \frac{1}{n} \sum_{i=\delta m+1}^n \mu_{i:n} \quad (24)$$

$$\approx \phi(\Phi^{-1}(1 - \delta\rho)). \quad (25)$$

In this case, the maximum of I_r is $\phi(0) = 1/\sqrt{2\pi} = 0.3989$, which is a fairly low value, but it is not negligible. When the migration policy is to replace individuals randomly in the target deme, there is no differential between the average fitness of the individuals that survive and the average fitness of the entire deme. Therefore, in this case $I_r = 0$.

The overall selection intensity caused by migration is simply

$$I_m = I_e + I_r. \quad (26)$$

To predict the number of generations until convergence, we should add I_m to the intensity from the selection method used to select the parents of the next generation in each deme (given in table I). The total intensity is then used in equation 10.

4.2. COMPARING THE MIGRATION POLICIES

The simple analysis of the takeover times in section 3 showed the trends in selection pressure that may be expected from the migration policies. This section confirms the previous observations and provides reference tables for frequently used configurations of parallel EAs.

Figure 2 presents plots of (the approximations of) I_m for topologies with different degrees and varying the migration rate². The maximum migration rate in the case when the best individuals migrate and replace the worst at the target is $\rho^* = 1/(\delta+1)$, where I_m reaches its maximum. At this migration rate, the migrants replace all but the $n/(\delta+1)$ best individuals already present in the deme. In the case when the best migrants replace random individuals, the maximum of I occurs at $\rho = 0.5$, but the highest migration rate that makes sense to use is $\rho^* = 1/\delta$ (for $\delta > 1$), because there are no more than $\delta\rho^*$ individuals in a deme. Finally, when random emigrants replace the worst individuals, the maximum of I_m is at $\rho^* = 1/(2\delta)$. In this case, the average fitness of the migrants is the same as the average fitness of the populations $\bar{f}_{mig}^t = \bar{f}^t$, and as long as the migrants replace the lowest half of the individuals in the receiving deme, the selection differential $s_r^t = \bar{f}_{sur}^t - \bar{f}^t$ will be positive. Beyond that point, the average fitness of the survivors would decrease.

From the plots it is easy to see that the migration policy with the highest intensity is when the best individuals migrate and replace the worst, followed closely by the case when the best migrants replace random individuals. The difference between these two policies is not as large as the difference with the policy where migrants are selected randomly, as predicted from the takeover time calculations.

² $\Phi^{-1}(x)$ in equations 21 and 25 was calculated numerically using Mathematica 3.0 as $\sqrt{2}\text{InverseErf}[0, 2x-1]$

Table II. I_m when the *best* individuals migrate and replace the *worst* individuals at the receiving deme.

ρ	0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.5
$\delta = 1$	0.053	0.206	0.350	0.466	0.559	0.635	0.695	0.740	0.772	0.791	0.797
$\delta = 2$	0.101	0.381	0.630	0.814	0.946	1.034	1.081				
$\delta = 3$	0.147	0.542	0.874	1.095	1.226	1.271					
$\delta = 4$	0.192	0.692	1.088	1.318	1.399						
$\delta = 5$	0.236	0.833	1.276	1.483							

Table III. I_m when the *best* individuals migrate and replace *random* individuals at the receiving deme.

ρ	0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.5
$\delta = 1$	0.026	0.103	0.175	0.233	0.279	0.317	0.347	0.370	0.386	0.395	0.398
$\delta = 2$	0.053	0.206	0.350	0.466	0.559	0.635	0.695	0.740	0.772	0.791	0.797
$\delta = 3$	0.079	0.309	0.526	0.699	0.839	0.953	1.043				
$\delta = 4$	0.106	0.412	0.701	0.932	1.119	1.271					
$\delta = 5$	0.133	0.515	0.877	1.165	1.399						

Table IV. I_m when *random* individuals migrate and replace the *worst* individuals at the receiving deme.

ρ	0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.5
$\delta = 1$	0.026	0.103	0.175	0.233	0.279	0.317	0.347	0.370	0.386	0.395	0.398
$\delta = 2$	0.048	0.175	0.279	0.347	0.386	0.398	0.386				
$\delta = 3$	0.068	0.233	0.347	0.395	0.386	0.317					
$\delta = 4$	0.086	0.279	0.386	0.386	0.279						
$\delta = 5$	0.103	0.317	0.398	0.317							

Table V. Selection intensity for different tournament sizes ($\mu_{s:s}$)

s	2	3	4	5	6	7	8	16	32
I	0.564	0.846	1.029	1.162	1.267	1.352	1.423	1.765	2.069

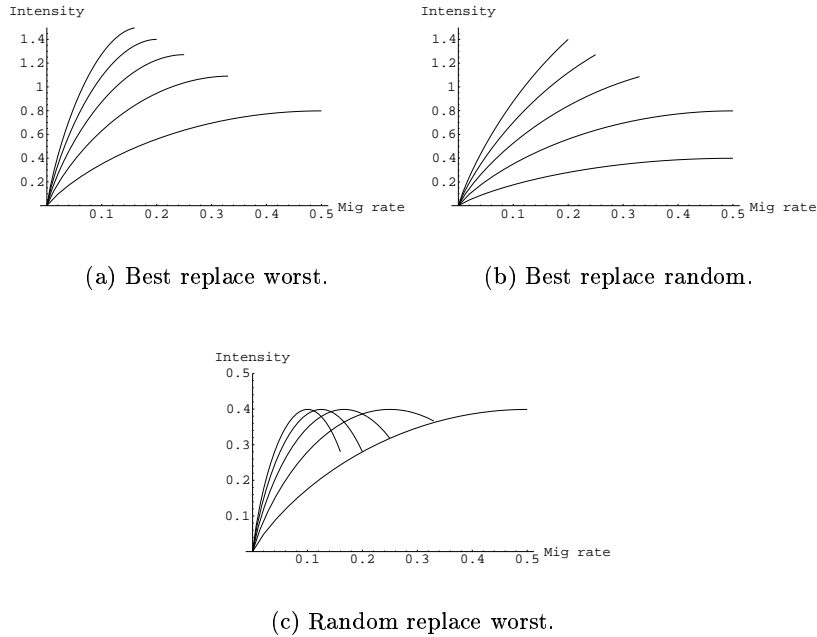


Figure 2. Selection intensity for different migration policies varying the number of neighbors ($\delta=1,2,\dots,5$ from bottom to top) and the migration rate.

Tables II, III, and IV have the values of I_m for common configurations. Observe that the magnitudes of I_m are fairly large. For example, in the case where the best migrants replace the worst individuals, a hypercube of degree $\delta = 3$ and using only 5% migration produces approximately the same selection intensity as tournament selection with $s = 3$ (see table V). Further comparisons between the tables of I_m and table V suggest that migration alone is capable of producing significant selection pressures and can cause the populations to converge fast.

Notice that when the best individuals migrate, the relation between δ and ρ is not linear. The selection intensity produced by two neighbors using $\rho = 0.05$ is greater than the selection intensity caused by one neighbor sending a fraction $\rho = 0.10$ of its individuals.

As an aside, note that it is possible to *reduce* the selection pressure with migration. The largest reduction would result from selecting the worst individuals to migrate and replace the best individuals of the receiving deme. Selecting the worst individuals to migrate and replacing randomly in the target deme would produce an intermediate reduction of selection pressure. The smallest reduction would be caused

by selecting emigrants at random and replacing the best individuals. Calculations similar to those in the previous section would quantify the decrease in selection intensity. But, why would we want to reduce the selection intensity with migration? The answer is the same as in a serial GA: to slow down convergence so that the variation operators have enough time to create new solutions. This may be particularly important when low crossover rates are used or when the demes are too small.

5. Experiments

This section presents experimental evidence that verifies the accuracy of the predictions of the previous section. Experiments are performed with the four migration policies, and the results shown are the average of 20 independent runs for each parameter setting.

All experiments use a 500-bit OneMax function, and the populations are initialized randomly (on average $P_0 = 0.5$). Each deme is a generational GA with $n = 100$ individuals, which is sufficient to ensure convergence to the optimum in all cases. The GAs use pairwise tournament selection ($s = 2$, $I = 0.5642$), uniform crossover with probability 1.0, and no mutation. The experiments vary the migration rate and the degree of the topology. Migration occurs every generation.

5.1. BEST MIGRANTS REPLACE WORST INDIVIDUALS

The first set of experiments uses equation 9, which predicts the number of bits correct over time, to assess the accuracy of the calculations of I . Figure 3 presents the results of experiments using a fully connected topology with two demes ($\delta = 1$). Note that the accuracy of the predictions decreases slightly as higher migration rates are used. A possible explanation for the small discrepancies is that it is likely that the migrants are different from the individuals already present in a deme (although their average fitness is the same). The increased diversity would require additional mixing (crossover) of alleles to produce a distribution more similar to a binomial. The problem is aggravated as longer strings and higher migration rates are used. The predictions should be much more accurate for algorithms such as PBIL (Baluja, 1994), UMDA (Mühlenbein and Paaß, 1996), or the compact GA (Harik et al., 1998), which treat each bit independently and do not suffer from the inadequate mixing problem described here. However, the predictions are adequate for the purposes of this paper, and therefore the effect of increased diversity due to migration is ignored here, but should be the subject of future research.

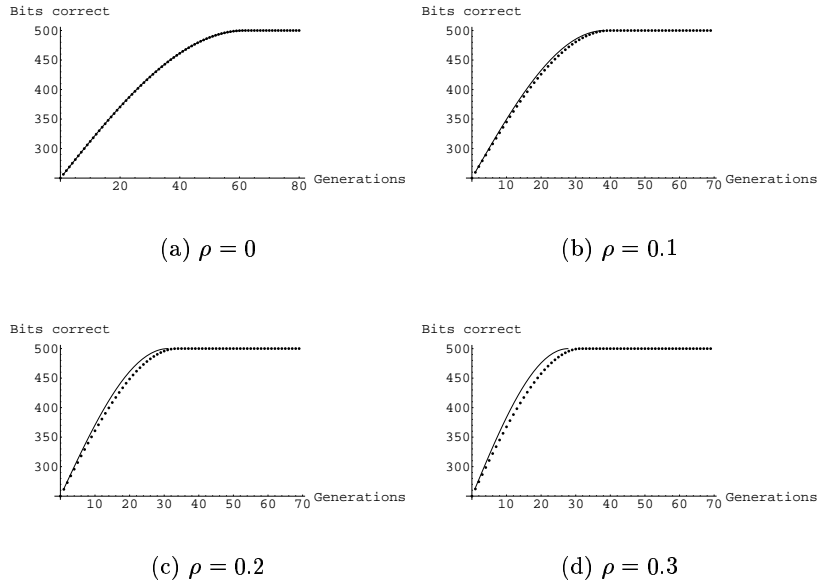


Figure 3. Bits correct in a topology with one neighbor using different migration rates. The line is the prediction from equation 9 and the dots are the experimental results.

The next set of experiments is designed to verify the prediction of the number of generations until convergence (eq. 10). In this migration policy, both the choices of migrants and replacements increase the selection pressure, and therefore I_m is calculated by adding equations 21 and 25. Then, the selection intensity from pairwise tournament selection is added, and the result is used as I in equation 10. The theoretical predictions and the experimental results are presented in figure 4.

Experiments with other topologies of the same degree show no difference. In particular, experiments with 8 demes connected as uni- and bi-directional rings ($\delta = 1$ and 2, respectively) and hypercubes of degree $\delta = 3$ and 4 (16 demes) yield identical results as those with fully-connected topologies in figures 3 and 4. For this reason, in the remainder we experiment only with fully connected topologies.

5.2. BEST MIGRANTS REPLACE RANDOM INDIVIDUALS

For this migration policy, I_e is given by equation 21 and $I_r = 0$. The results are presented in figure 5. We can observe that the generations until convergence are only slightly fewer than in the previous case, where the replacements were chosen according to their fitness. This observation

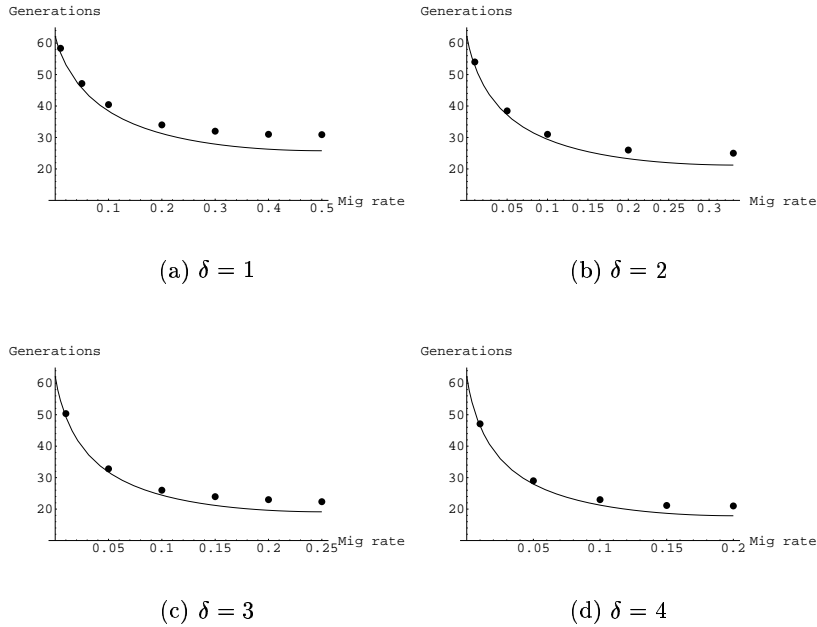


Figure 4. Number of generations until convergence when the *best* individuals are selected to migrate and replace the *worst*. The line plots the predictions (eq. 10) and the dots are the experimental results.

supports the hypothesis raised by the analysis of takeover times that indicated that the major component in the increase of selection pressure is the selection of emigrants, not the selection of replacements.

5.3. RANDOM MIGRANTS REPLACE WORST INDIVIDUALS

In this migration policy, there is no selection pressure caused from selecting random emigrants ($I_e = 0$), but replacing the worst individuals in the target deme causes the intensity to increase. I_r is given by equation 25.

We can observe in figure 6 that since the additional selection pressure caused by this migration policy is not very strong, the generations until convergence do not decrease as much as in the previous cases. We used migration rates higher than the maximum $\rho^* = 1/(2\delta)$ rate that makes sense in this case to be consistent with the previous experiments and to illustrate how the selection pressure decreases after this point.

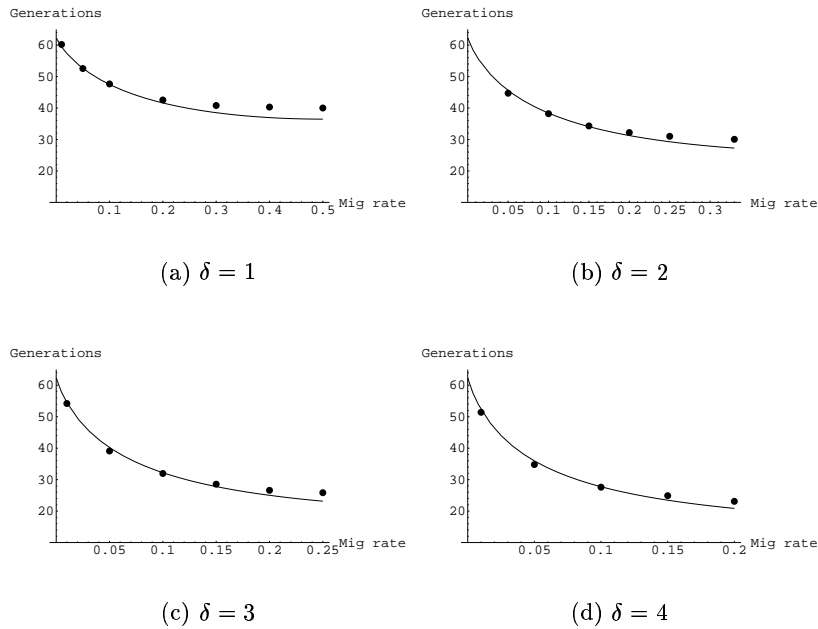


Figure 5. Number of generations until convergence when the *best* individuals are selected to migrate and replace *random* individuals. The line plots the predictions (eq. 10) and the dots are the experimental results.

6. Superlinear Speedups in EAs

In the EA community there has been some controversy about claims of superlinear speedups. The primary reason to suspect these claims is a common argument used to dismiss superlinearity in general: if all the tasks of a parallel program are executed by several threads on a single processor, the total execution time cannot be less than the execution time of a serial program that performs the *same* computations. As Punch (1998) points out, the key assumption is that that the serial and parallel programs execute the exact same tasks. In this view, the only possible explanation of superlinear speedups is that the parallel EA somehow executes less work than the serial EA. The argument of this paper is that in many cases the reduction of computations is caused by the increased selection pressure of migration.

To illustrate how the different convergence times affect the speedups of parallel GAs consider an example with the 500-bit OneMax problem used in the previous section. A simple GA with a single population of $n = 100$ individuals and the same parameters as before reached the

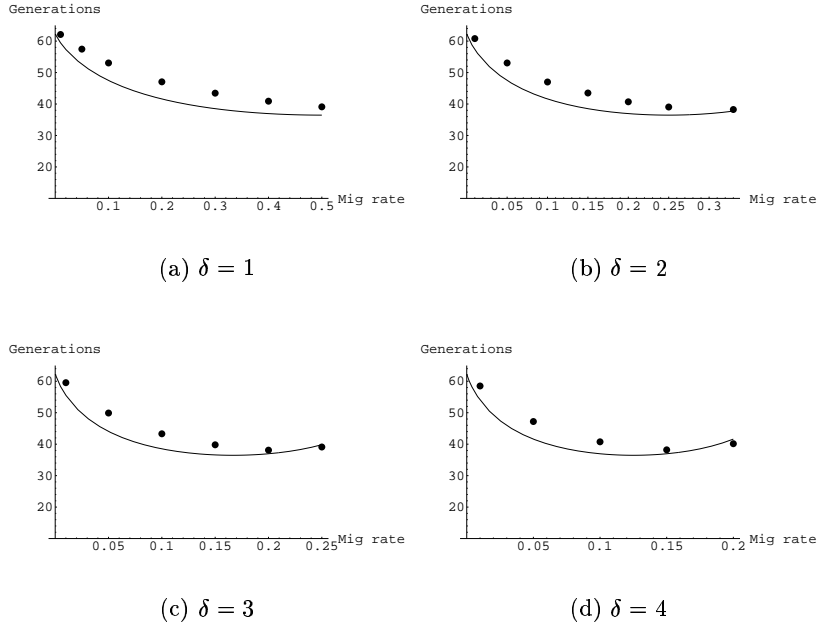


Figure 6. Number of generations until convergence when *random* individuals are selected to migrate and replace the *worst*. The line plots the predictions (eq. 10) and the dots are the experimental results.

global solution in all of 20 independent trials. We verified empirically that this population size was the smallest that consistently reached the global optimum. The execution time of the serial GA can be estimated as $T_s = nG_sT_f$, where G_s is the number of generations until convergence and T_f is the function evaluation time. With $l = 500$, $P_0 = 0.5$, and using pairwise tournaments ($I = 0.5642$), equation 10 gives $G_s = 62$. For the purposes of the example, let $T_f = 1$ unit of time, and therefore $T_s = 6200$.

For the parallel case, consider $r = 4$ demes with $n = 25$ individuals each. The demes are connected by a bi-directional ring ($\delta = 2$) and use a migration rate of $\rho = 0.05$. We validated empirically that this configuration reaches the optimum in all of 20 independent runs, and that this was the smallest deme size to do so. The execution time in the parallel case is $T_p = nG_pT_f + T_c$, where T_c is the time used in communications. It depends on δ and ρ , and for our example we will

use a large value $T_c = 100^3$. The value of $I_m = 0.381$ may be found in table II. Substituting $I = 0.5642 + 0.381$ in equation 10 gives $G_p = 37$. With these values $T_p = 1025$, and the ratio $\frac{T_s}{T_p} = 6.05$, which even with the conservative values used here is much higher than the ideal speedup of 4.

We should not conclude that all claims of superlinear speedups are caused by an increase of selection pressure due to migration. Other possible causes are implementation particulars (for example, the smaller demes may fit completely in the processors' caches, which was the explanation that Belding (1995) gave for his results) or inappropriate sizing of populations (such that convergence to solutions of the same quality is not guaranteed).

7. Summary and Future Research Directions

The choice of migrants and the replacement of individuals are not often considered important parameters of parallel EAs. However, this paper showed that choosing migrants or replacements according to their fitness increases the selection pressure and may cause the algorithm to converge significantly faster.

The migration policy that causes the greatest reduction in work is to choose both the migrants and the replacements according to their fitness, which is also the most common policy. The results of the paper also indicate that the selection pressure increases monotonically with higher migration rates. The faster convergence may result in a reduction of the total computational work, and may explain some claims of superlinear speedups in parallel EAs.

These conclusions were obtained with two different methods. The first method is extremely simple, but produced valuable qualitative observations that prompted further investigation with more complex tools. Besides of the results that were obtained, this paper illustrates a methodology may be used to explore other areas of EAs.

As was mentioned in section 5.1, migration also introduces diversity into a deme. Although it seems that the selection intensity largely determines the convergence speed, in future research the effect of the increased diversity may be quantified and incorporated into the calculations to obtain more accurate predictions.

Another open area for future research is to calculate the selection intensity of different selection schemes used in fine-grained parallel EAs.

³ Notice that the ratio of T_c/T_f is very high. In most practical applications of parallel GAs this ratio is much lower than one. We are being extremely conservative here.

There is some important work in this area that is related with takeover times (e.g., (Sarma and De Jong, 1996; Sarma and De Jong, 1997)), but calculations of the selection intensity are still nonexistent. These would permit to compare directly the selection pressure across different models of parallel EAs.

Acknowledgments

I wish to thank Prof. David E. Goldberg for his guidance during the research reported here. I was partially supported by a Fulbright-García Robles Fellowship.

The work was sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number F49620-97-1-0050. Research funding for this project was also provided by a grant from the US Army Research Laboratory Program, Cooperative Agreement DAAL01-96-2-003. The US Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies and endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the US Government.

References

- Bäck, T.: 1994, 'Selective pressure in evolutionary algorithms: A characterization of selection mechanisms'. In: *Proceedings of the First IEEE Conference on Evolutionary Computation*, Vol. 1. Piscataway, NJ, pp. 57–62.
- Bäck, T.: 1995, 'Generalized convergence models for tournament- and (μ, λ) -selection'. in (Eschelman, 1995), pp. 2–8.
- Baker, J. E.: 1985, 'Adaptive Selection Methods for Genetic Algorithms'. In: J. J. Grefenstette (ed.): *Proceedings of an International Conference on Genetic Algorithms and Their Applications*. Hillsdale, NJ, pp. 101–111.
- Baker, J. E.: 1987, 'Reducing Bias and Inefficiency in the Selection Algorithm'. in (Grefenstette, 1987), pp. 14–21.
- Baluja, S.: 1994, 'Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning'. Tech. Rep. No. CMU-CS-94-163, Carnegie Mellon University, Pittsburgh, PA.
- Belding, T. C.: 1995, 'The distributed genetic algorithm revisited'. in (Eschelman, 1995), pp. 114–121.
- Blickle, T. and L. Thiele: 1996, 'A comparison of selection schemes used in evolutionary algorithms'. *Evolutionary Computation* 4(4), 361–394.
- Booker, L. B.: 1982, 'Intelligent Behavior as an Adaptation to the Task Environment'. Unpublished doctoral dissertation, The University of Michigan. (University Microfilms No. 8214966).

- Brindle, A.: 1981, 'Genetic algorithms for function optimization'. Unpublished doctoral dissertation, University of Alberta, Edmonton, Canada.
- Burrows, P.: 1972, 'Expected selection differentials for directional selection'. *Biometrics* **23**, 1091–1100.
- Cantú-Paz, E.: 1999, 'Migration Policies and Takeover Times in Parallel Genetic Algorithms'. In: W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith (eds.): *GECCO-99: Proceedings of the 1999 Genetic and Evolutionary Computation Conference*. San Francisco, CA, p. 775.
- Cantú-Paz, E. and D. E. Goldberg: 1999, 'Parallel genetic algorithms: theory and practice'. *Computer Methods in Applied Mechanics and Engineering*. In press.
- Eschelman, L. (ed.): 1995, 'Proceedings of the Sixth International Conference on Genetic Algorithms'. San Francisco, CA, Morgan Kaufmann.
- Goldberg, D. E. and K. Deb: 1991, 'A comparative analysis of selection schemes used in genetic algorithms'. *Foundations of Genetic Algorithms* **1**, 69–93. (Also TCGA Report 90007).
- Goldberg, D. E., K. Deb, and D. Thierens: 1993, 'Toward a better understanding of mixing in genetic algorithms'. *Journal of the Society of Instrument and Control Engineers* **32**(1), 10–16.
- Grefenstette, J. J.: 1981, 'Parallel adaptive algorithms for function optimization'. Tech. Rep. No. CS-81-19, Vanderbilt University, Computer Science Department, Nashville, TN.
- Grefenstette, J. J. (ed.): 1987, 'Proceedings of the Second International Conference on Genetic Algorithms'. Hillsdale, NJ, Lawrence Erlbaum Associates.
- Grosso, P. B.: 1985, 'Computer simulations of genetic adaptation: Parallel subcomponent interaction in a multilocus model'. Unpublished doctoral dissertation, The University of Michigan. (University Microfilms No. 8520908).
- Hancock, P. J. B.: 1997, 'Selection: a comparison of selection mechanisms'. In: T. Bäck, D. B. Fogel, and Z. Michalewicz (eds.): *Handbook of Evolutionary Computation*. Bristol and New York: Institute of Physics Publishing and Oxford University Press, pp. C2.8:1–C2.8:11.
- Harik, G. R., F. G. Lobo, and D. E. Goldberg: 1998, 'The compact genetic algorithm'. In: I. of Electrical and E. Engineers (eds.): *Proceedings of 1998 IEEE International Conference on Evolutionary Computation*. Piscataway, NJ, pp. 523–528.
- Harter, H. L.: 1970, *Order Statistics and Their Use in Testing and Estimation*. Washington, D.C.: U.S. Government Printing Office.
- Holland, J. H.: 1975, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press.
- Lin, S.-C., E. D. Goodman, and W. F. Punch III: 1997, 'Investigating parallel genetic algorithms on job shop scheduling problems'. In: P. J. Angeline, R. G. Reynolds, J. R. McDonnell, and R. Eberhart (eds.): *Evolutionary Programming VI*. Berlin, pp. 383–393.
- Miller, B. L. and D. E. Goldberg: 1995, 'Genetic algorithms, tournament selection, and the effects of noise'. *Complex Systems* **9**(3), 193–212.
- Miller, B. L. and D. E. Goldberg: 1996, 'Genetic algorithms, selection schemes, and the varying effects of noise'. *Evolutionary Computation* **4**(2), 113–131.
- Mühlenbein, H.: 1991, 'Evolution in time and space-The parallel genetic algorithm'. In: G. J. E. Rawlins (ed.): *Foundations of Genetic Algorithms*. San Mateo, CA, pp. 316–337.
- Mühlenbein, H. and G. Paaß: 1996, 'From recombination of genes to the estimation of distributions I. Binary parameters'. in (Voigt et al., 1996), pp. 178–187.

- Mühlenbein, H. and D. Schlierkamp-Voosen: 1993, 'Predictive models for the breeder genetic algorithm: I. Continuous parameter optimization'. *Evolutionary Computation* **1**(1), 25–49.
- Punch, W. F.: 1998, 'How effective are multiple programs in genetic programming'. In: J. R. Koza, W. Banzhaf, K. Chellapilla, K. Deb, M. Dorigo, D. B. Fogel, M. H. Garzon, D. E. Goldberg, H. Iba, and R. L. Riolo (eds.): *Genetic Programming 98*. San Francisco, pp. 308–313.
- Sarma, J. and K. De Jong: 1996, 'An analysis of the effects of neighborhood size and shape on local selection algorithms'. in (Voigt et al., 1996), pp. 236–244.
- Sarma, J. and K. De Jong: 1997, 'An analysis of local selection algorithms in a spatially structured evolutionary algorithm'. In: T. Bäck (ed.): *Proceedings of the Seventh International Conference on Genetic Algorithms*. San Francisco, pp. 181–187.
- Schaffer, J. D. (ed.): 1989, 'Proceedings of the Third International Conference on Genetic Algorithms'. San Mateo, CA:, Morgan Kaufmann.
- Schwefel, H.: 1981, *Numerical Optimization of Computer Models*. Chichester: John Wiley and Sons.
- Syswerda, G.: 1989, 'Uniform crossover in genetic algorithms'. in (Schaffer, 1989), pp. 2–9.
- Tanese, R.: 1987, 'Parallel genetic algorithm for a hypercube'. in (Grefenstette, 1987), pp. 177–183.
- Tanese, R.: 1989, 'Distributed genetic algorithms'. in (Schaffer, 1989), pp. 434–439.
- Thierens, D. and D. E. Goldberg: 1993, 'Mixing in genetic algorithms'. In: S. Forrest (ed.): *Proceedings of the Fifth International Conference on Genetic Algorithms*. San Mateo, CA, pp. 38–45.
- Thierens, D. and D. E. Goldberg: 1994, 'Convergence models of genetic algorithm selection schemes'. In: Y. Davidor, H.-P. Schwefel, and R. Männer (eds.): *Parallel Problem Solving from Nature, PPSN III*. Berlin, pp. 119–129.
- Voigt, H.-M., W. Ebeling, I. Rechenberg, and H.-P. Schwefel (eds.): 1996, 'Parallel Problem Solving from Nature, PPSN IV'. Berlin:, Springer-Verlag.
- Whitley, D.: 1993, 'An executable model of a simple genetic algorithm'. In: L. D. Whitley (ed.): *Foundations of Genetic Algorithms 2*. San Mateo, CA, pp. 45–62.
- Whitley, D., S. Rana, and R. B. Heckendorn: 1999, 'Exploiting Separability in Search: The Island Model Genetic Algorithm'. *Journal of Computing and Information Technology*. forthcoming.