The science of breeding and its application to the breeder genetic algorithm BGA

Heinz Mühlenbein, Dirk Schlierkamp-Voosen [*]
GMD Schloss Birlinghoven
D-53754 Sankt Augustin
e-mail: muehlen@gmd.de

Abstract

The Breeder Genetic Algorithm BGA models artificial selection as performed by human breeders. The science of breeding is based on advanced statistical methods. In this paper a connection between genetic algorithm theory and the science of breeding is made. We show how the response to selection equation and the concept of heritability can be applied to predict the behavior of the BGA. Selection, recombination and mutation are analyzed within this framework. It is shown that recombination and mutation are complementary search operators. The theoretical results are obtained under the assumption of additive gene effects. For general fitness landscapes regression techniques for estimating the heritability are used to analyze and control the BGA. The method of decomposing the genetic variance into an additive and a nonadditive part connects the case of additive fitness functions with the general case.

1 Introduction

Evolution of natural organisms is based on three major components - reproduction, variation and selection. Some reproductions of natural organisms occur with "failures" called mutations. A more systematic variation of the genetic material happens in sexual reproduction. Each parent contributes half of its genetic material to the offspring. This method of variation is called recombination. The offspring will be identical to the parents if the parents are genetically equal.

Variation is necessary to allow selection to work. Selection in nature is very difficult to define precisely. The term was introduced by Darwin (1859) very informally. "*The preservation of favourable variations and the rejection of injurious variations, I call Natural Selection.*" But how can an observer predict which are the favorable variations? The favorable variations are the variations which are preserved! The variations can only be judged after they have competed in the "struggle for life." Natural selection is no independent force of nature, it is the result of the competition of natural organisms for resources.

In contrast, in the science of breeding the above problem does not exist. The selection is done by human breeders. Their strategies are based on the assumption that mating two individuals with high fitness more likely produces an offspring of high fitness than

[*] appeared in: Evolutionary Computation 1(4):335-360, 1994

two randomly mating individuals. The *Breeder Genetic Algorithm* **BGA** introduced in (Mühlenbein & Schlierkamp-Voosen, 1993) is based on the science of breeding. The science is part of applied statistics. A major component is the parent-offspring correlation and the heritability coefficient.

There is one major difference between breeding of natural organisms and the breeder genetic algorithm. The human breeder has no influence on the genetic operators mutation and recombination. This is done by nature. But evolutionary algorithms have to simulate the genetic operators. Therefore the genetic operators as well as their frequency of occurrence can be implemented to optimize the breeding process. For instance, the mutation rate used by the breeder genetic algorithm is larger than the one found in nature. With biotechnology breeders could also potentially begin to control mutation and recombination. It is interesting to note that as early as 1930 breeders had been very enthusiastic about the possibility of increasing the mutation rate by radiation. But experiments later showed that this brute force method is not effective for breeding animals. The percentage of lethal mutation was too high.

In this paper we deal initially with a rather simplified model. We assume additive gene contributions and uniform crossover. Nevertheless five parameters are needed to describe the initial state of the population and the selection process. The necessary parameters are

- the population size N
- the initial frequency of the desirable allele p_0
- the number of loci n
- the mutation rate m
- the intensity of selection I

For this model we will compute the expected number of generations until convergence. It would be futile to investigate the model with all five parameters variable. Therefore we will investigate the model with one or more parameters fixed. The outline of the paper is as follows.

First we will investigate evolution without selection, also called *genetic drift* ($I = 0$). If there is no mutation the population will converge to a unique genotype. In section three we will analyze selection and recombination in large populations. The analysis is based on the *response to selection equation* and on the concept of *heritability*. Then selection and recombination are analyzed in small populations by simulations. In sections five and six mutation in small and large populations is investigated. In section seven the major results are summarized and discussed.

The above theory gives a clear picture about the behavior of the major evolutionary components. For the breeder genetic algorithm this theory plays the same role as the ideal gas theory for classical thermodynamics. The "ideal gas" in evolutionary algorithms are simple additive fitness functions.

The theory will be extended in section eight to more general fitness functions. The key concept is estimating the heritability by a regression of offspring to parent. This result will be used to estimate the heritability by taking a genetic chance model into account. This method allows the variance of the fitness of the population to be decomposed into an additive genetic component and epistatic interaction components. An important result of this approach is the proof that only additive fitness functions have a heritability of one. All other functions have a heritability less than one.

In the last section we will apply the theory to understand and control the BGA. In a full BGA all evolutionary components are used. In addition very complex fitness landscapes have to be searched. We will show that the regression estimates can be used to predict the heritability and to effectively control the BGA.

Some of the results presented in this paper are also of interest for population genetics. Our models are restricted to haploid organisms. But in this area our models and equations are sometimes more precise than the ones used in population genetics. Examples are the analysis of genetic drift and the analysis of the genetic variance. For a recent survey about predicting the response to selection in livestock productions see (Verrier, Colleau & Foulley, 1991).

2 Evolution without selection - genetic drift

It has been known in population genetics for quite some time that a finite population converges to a single genotype, even if selection is not applied. The mutation rate is assumed to be negligible. The fixation of the population is a result of its finite size. This effect has been called *genetic drift* by Wright (1932). The importance of genetic drift for explaining evolution in nature has been emphasized by Kimura (1983). He developed a neutral theory of molecular evolution, claiming that natural selection is not as important for evolution as previously surmized. Kimura used a very complex diffusion equation approach to quantify genetic drift (Crow & Kimura, 1970). We will generalize his results. Two chance models will be distinguished

1. no selection, no recombination
2. no selection, but with recombination

The first model is just sampling with replacement. The second model is an adaptation of Mendel's genetic chance model to haploid organisms. For the analysis of genetic effects the following cases will be distinguished if necessary:

- one gene with two alleles
- n genes each with two alleles
- n genes with an infinite number of alleles

The last case roughly models the genetic representation used by the BGA for continuous functions of n variables. In all cases, recombination is done by randomly choosing an allele from one of the parents. For binary representations this recombination scheme is called *uniform crossover* (Syswerda, 1989).

The next three theorems have been derived in (Asoh & Mühlenbein, 1994b). The proofs are based on a Markov chain analysis for one gene with two alleles. The formulas have been obtained by numerically fitting the data.

Theorem 1 *Let there be a gene with two alleles. Let half of the initial population have allele* 0*, the other allele* 1*. Then in a randomly mating population of size N without* *mutation and recombination, the expected number of generations until equilibrium GEN_e is given by*

$$E(GEN_e) \approx 1.4 \cdot N \tag{1}$$

If the number of genes or the number of alleles is very large, GEN_e is only slightly larger. This is shown in the next theorem.

Theorem 2 *Let the number of genes or alleles be large enough, that the genotypes of the initial population are all different from each other. Then in a randomly mating population of size N without mutation and recombination, the expected number of generations until equilibrium GEN_e is approximately*

$$E(GEN_e) \approx 2 \cdot N \tag{2}$$

In table 1 numerical results from simulations are given. They are averages over 10,000 runs. Note the very large standard deviation SD.

N	8	16	32	6 4	128	256	512
GEN_e	13.6	29.4	60.3	128.0	245.2	546.0	1131.1
SD	7.8	16.7	33.5	72.1	121.1	294.9	736.5

Table 1: Gen_e for a large number of genes

The theorems are in agreement with the results of Crow and Kimura (1970). They obtained for diploid chromosomes twice as large values, i.e $GEN_e = 2.8N$ and $GEN_e = 4N$.

The next theorem gives the convergence time if recombination is applied. It is restricted to binary representations. This theorem is new.

Theorem 3 *Let each gene have two alleles. Let the size of the chromosome be n, the size of the population be N. Let the initial population be randomly generated. Then for a randomly mating population with no selection, but with uniform crossover, the expected number of generations until equilibrium is approximately*

$$E(GEN_e) \approx 1.4 \cdot N \cdot (0.5ln(n) + 1)^{1.1} \tag{3}$$

Table 2 gives some results of BGA simulations. One clearly observes that GEN_e increases linearly with the popsize N and only logarithmically with the size of the problem n. This result shows that recombination is not able to substantially reduce the influence of genetic drift. We will later show that genetic drift is indeed an important factor if small selection intensities are used.

The results for an infinite number of alleles case are qualitatively similar. To summarize some results obtained by simulations, they show that GEN_e scales as $N \cdot ln(n)$, similar to the binary case. It seems that the value of GEN_e for an infinite number of alleles is about the value of GEN_e for the binary case with twice as many genes. The popsizes are held equal.

In the next section we will analyze selection and recombination in large populations.

	N		
n	16	32	64
32	67.1	131.0	261.9
64	77.6	160.2	334.2
512	107.7	224.0	475.4
1024	123.6	247.6	504.3
4096	141.6	289.0	

Table 2: GEN_e for different n and $N = 16, 32, 64$ with recombination (two alleles)

3 Response to selection

In this section we summarize the theory presented in (Mühlenbein & Schlierkamp-Voosen(1993b). The change produced by selection that mainly interests the breeder is the *response to selection*, which is symbolized by R. R is defined as the difference between the population mean fitness of generation $t+1$ and the population mean of generation t. $R(t)$ measures the expected progress of the population.

$$R(t) = M(t+1) - M(t) \tag{4}$$

where $M(t)$ denotes the average of the population at generation t. Breeders measure the selection with the *selection differential*, which is symbolized by S. It is defined as the difference between the mean fitness of the selected parents $M_s(t)$ and the mean fitness of the population.

$$S(t) = M_s(t) - M(t) \tag{5}$$

Breeders often use *truncation selection* or *mass selection*. In truncation selection with threshold T, the T % best individuals will be selected as parents. T is normally chosen in the range 10% to 50%.

The prediction of the response to selection starts with

$$R(t) = b_t \cdot S(t) \tag{6}$$

b_t is called *realized heritability* in quantitative genetics. The breeder either measures b_t in previous generations or estimates b_t by different methods. Two popular methods based on the regression of parents to offspring will be explained later. It is normally assumed that b_t is constant for a certain number of generations. This leads to

$$R(t) = b \cdot S(t) \tag{7}$$

There is no genetics involved in this equation. It is simply an extrapolation from direct observation. The prediction of just one generation is only half the story. The breeder (and the GA user) would like to predict the cumulative response R_s for s generations of his breeding scheme.

$$R_s = \sum_{t=1}^{s} R(t) = b \sum_{t=1}^{s} S(t) \tag{8}$$

The response to selection is the product of the heritability and the selection differential. For predicting the response to selection b and the selection differential have to be estimated.

If the fitness values are normal distributed, the selection differential $S(t)$ in truncation selection is approximately given by

$$S(t) = I \cdot \sigma_p(t) \tag{9}$$

where σ_p is the phenotypical standard deviation. I is called the *selection intensity.* The formula is a feature of the normal distribution. A derivation can be found in (Bulmer, 1980).

The science of artificial selection consists of estimating b and $\sigma_p(t)$. We just cite the following theorem (Mühlenbein & Schlierkamp-Voosen, 1993b). It was proven for the ONEMAX function under the assumption that $\sigma_p(t)$ has a binomial distribution

$$\sigma_p(t) = \sqrt{n \cdot p(t) \cdot (1 - p(t))}$$

$p(t)$ is the probability of the advantageous allele in the population at generation t.

Theorem 4 *Let the breeder genetic algorithm be run with uniform crossover. If the population is large enough that it converges to the optimum and if the selection intensity I is greater than* 0*, then the probability of the advantageous bit $p(t)$ is given for the $ONEMAX$ function by*

$$p(t) = 0.5 \cdot \left(1 + sin\left(\frac{I}{\sqrt{n}}t + arcsin(2p_0 - 1)\right)\right) \tag{10}$$

The number of generations needed until equilibrium is approximate

$$GEN_e = \left(\frac{\pi}{2} - arcsin(2p_0 - 1)\right) \cdot \frac{\sqrt{n}}{I} \tag{11}$$

$p_0 = p(0)$ *denotes the probability of the advantageous bit in the initial population.*

We next compare the analytical results with simulations. In figure 1 the mean fitness versus the number of generations is shown for three popsizes $N = 1024, 256,$ and 64. The selection intensity is $I = 0.8$, the size of the problem $n = 64$. The initial population was generated with $p_0 = 1/64$.

A closer look at the simulation results show that the phenotypic variance is slightly less than given by the binomial distribution. The empirical data is better fitted if the following estimate is used

$$\tilde{\sigma}_p(t) = \frac{\pi}{4.3}\sqrt{n \cdot p(t) \cdot (1 - p(t))} \tag{12}$$

Using this variance one obtains the equations

$$\tilde{R}(t) = \frac{\pi}{4.3} \cdot I \cdot \sqrt{n \cdot p(t)(1 - p(t))} \tag{13}$$

$$G\tilde{E}N_e = \frac{4.3}{\pi}\left(\frac{\pi}{2} - arcsin(2p_0 - 1)\right) \cdot \frac{\sqrt{n}}{I} \tag{14}$$

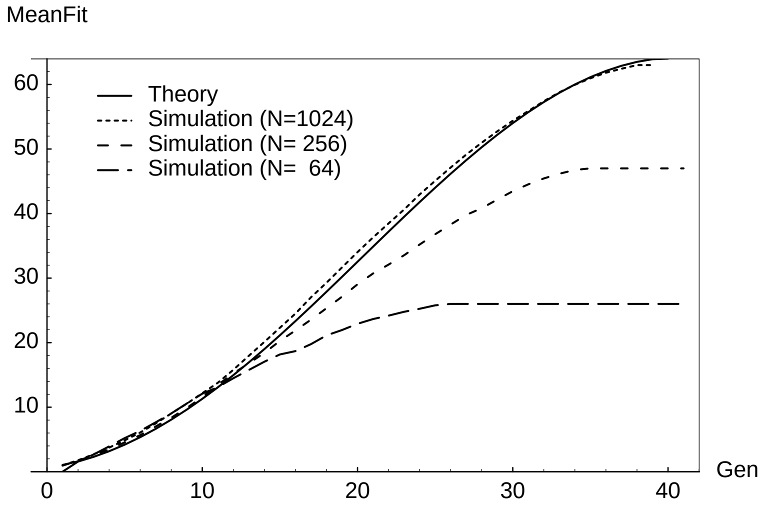

Figure 1: Mean fitness for various N ($T = 0.5, p_0 = 1/64$). $N = 64$ converges first.

The fit of equation 13 and the simulation run with $N = 1024$ is very good. For $N = 256$ and $N = 64$ the population does not converge to the optimum. These popsizes are less than the *critical popsize* $N^*(I, n, p_0)$. The critical popsize is defined to be the minimum popsize that the BGA converges with high probability to the optimum. The problem of determining the critical popsize will be discussed later.

We have not been able to prove a similar theorem for an infinite number of alleles. The difficulty lies in estimating the variance of the population. We will give some simulation results in the next section.

For proportionate selection which is used by the simple genetic algorithm we extend the theorem already proven in (Mühlenbein & Schlierkamp-Voosen, 1993b).

Theorem 5 *For a genetic algorithm using proportionate selection the selection differential is given by*

$$S(t) = \frac{\sigma_p^2(t)}{M(t)} \tag{15}$$

For the $ONEMAX$ function of size n the response to selection can be computed from

$$R(t) = 1 - p(t) \tag{16}$$

If the population is large enough, the number of generations until $p(t) = 1 - \epsilon$ is given for large n by

$$GEN_{1-\epsilon} \approx n \cdot ln \frac{1 - p_0}{\epsilon} \tag{17}$$

p_0 is the probability of the advantageous allele in the initial population.

Proof *We will only prove 17. For $ONEMAX(n)$ we have $R(t) = S(t)$. As before we approximate the variance by the variance of the binomial distribution*

$$\sigma_p^2(t) \approx np(t)(1-p(t)) \tag{18}$$

Because $M(t) = np(t)$, equation 16 is obtained. From $R(t) = n(p(t+1) - p(t))$ we get the difference equation

$$p(t+1) = \frac{1}{n} + (1 - \frac{1}{n})p(t) \tag{19}$$

This equation has the solution

$$p(t) = \frac{1}{n}\left(1 + (1-\frac{1}{n}) + \cdots + (1-\frac{1}{n})^{t-1}\right) + (1-\frac{1}{n})^t p_0$$

This equation can be simplified to

$$p(t) = 1 - (1-\frac{1}{n})^t(1-p_0)$$

By setting $p(GEN_{1-\epsilon}) = 1 - \epsilon$ equation 17 is easily obtained.

This theorem shows the problem of proportionate selection. It selects too weak if the population approaches the optimum.

Both theorems of this section assume large popsizes. In the next section we will analyze small populations by simulations.

4 Analysis of recombination in small populations

We have not been able to analytically estimate the time to convergence for arbitrary population size N. Therefore we will use simulations in this section. First we will explain the behavior of the recombination operator for the binary $ONEMAX(n)$ function. In figure 2 the number of generations GEN_e until equilibrium and the size of the population N are displayed. The problem size is 64. The initial population was randomly generated with probability $p_0 = 0.2$ of the advantageous allele. The data are averages over 100 runs.

The figure can be divided into two areas. The first area can be called the *saturation region*. The population size is large enough so that the population converges to the optimum value. In this area GEN_e is constant. It can be predicted from equation 14 . For small popsizes GEN_e increases with the size of the population. The popsizes are too small to reach the optimum. But the quality of the final solution gets better with increasing popsize. This is the major reason for the increase of GEN_e.

The two regions are separated by a line which is given by the *critical population size* N^*. It is the minimal population size so that the population converges to the optimum. N^* depends on the selection intensity I, the size of the problem and the initial population. The relation between N^* and I is especially difficult. N^* increases for small and large selection intensities I. The increase of N^* with decreasing I for $I \leq 0.6$ seems surprising. The explanation for this behavior is *genetic drift*. If the popsize is too small, the variation of the population will be substantially reduced by genetic drift. The population will not converge to the optimum. The problem of estimating N^* is very difficult because the transition from region 2 to the saturation region is very slow. In (Mühlenbein & Schlierkamp-Voosen, 1993b) it was conjectured that for $p_0 = 0.5$

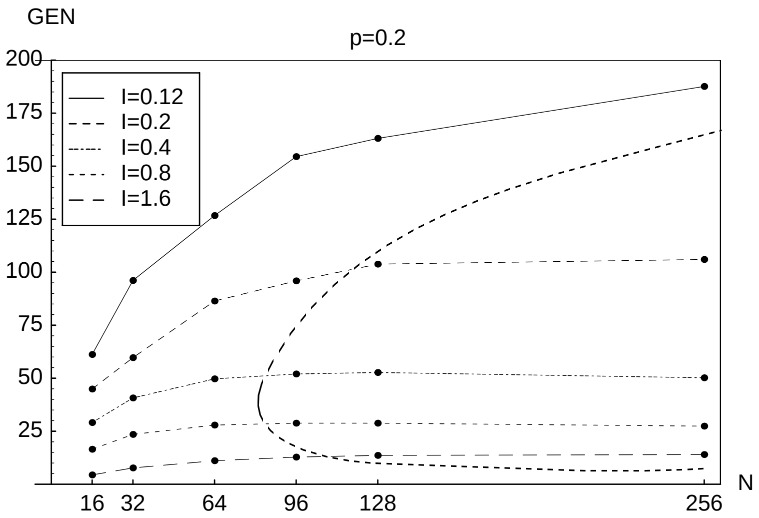

Figure 2: GEN_e vs population size N for $p_0 = 0.2$. The critical popsize N^* is indicated by a dashed line.

$$N^* \approx \sqrt{n} \cdot ln(n) \cdot f(I) \tag{20}$$

We have not yet made enough experiments to confirm the above formula. Very small selection ($I \approx 0$) defines the left boundary. This area is called *genetic drift.* It has already been investigated in section two. Here GEN_e increases at the rate $O(N \cdot ln(n))$

From GEN_e the number of function evaluations FE till convergence can be easily computed by

$$FE_e = N \cdot GEN_e$$

The minimum number FE_e^* is given by

$$FE_e^* = \min_I \{N^* \cdot GEN_e\}$$

For the $ONEMAX$ function FE_e^* is a flat minimum. The minimum is found at about $I = 1$. But FE_e^* does not differ much in the range $0.8 < I < 1.6$. This shows that the efficiency of the search does not crititically depend on the truncation threshold (Mühlenbein & Schlierkamp-Voosen, 1993b).

Slightly different results are obtained in the case of an infinite number of alleles. Here the critical popsize is infinite. Therefore a saturation area does not exist. The number of generations until convergence increases, but the quality of the final solution also improves. We take as example the optimization of the negative hypersphere in 32 dimensions.

$$F_0(x) = -\sum_i^n x_i^2$$

Simulations show that GEN_e for truncation thresholds of $T \geq 0.5$ increases only logarithmic in the size of the population N. This is surprisingly small. The major difference between binary alleles and an infinite number of alleles is shown in figure 3. For the simulations we used a BGA with discrete recombination. Five curves are displayed which show the increase of the average fitness for a single run. The mean fitness is displayed on a logarithmic scale. In all runs the mean fitness increases linearly till shortly before equilibrium. Therefore, mathematically, our recombination method has a *linear order of convergence.* The increase of the mean fitness is only dependent on the selection threshold. Severe selection leads to a larger slope, but the final fitness achieved with the same popsize is lower. Better final fitness values can be achieved by increasing the population size or decreasing the search domain. Note that decreasing the search domain from $[-1, 1]$ to $[-0.1, 0.1]$ gives better results than increasing the population size from 512 to 4000.

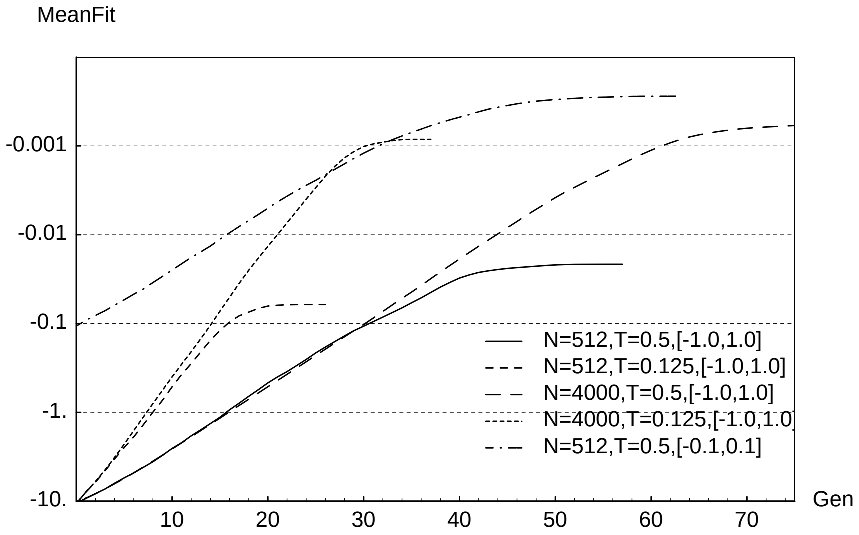

Figure 3: Mean fitness for thresholds $T = 0.5$ and 0.125 and $N = 512$ and 4000. Note that the mean fitness is displayed on a logarithmic scale. The increase of the mean fitness mainly depends on the truncation threshold. Severe selection leads to a larger gradient of the mean fitness, but the best fitness achieved is lower. Smaller populations converge faster, but also to a smaller fitness value.

For continuous functions more specific recombination methods can be applied which for instance arithmetically combine the two alleles of the parents. In the BGA we mainly use *line recombination* (Mühlenbein & Schlierkamp-Voosen, 1993b). In this method the offspring is randomly placed on the straight line which connects the two parent points. Simulations show that the behavior of line recombination is very similar to discrete recombination. For a given population size the number of generations until convergence is larger but the quality of the solution is also better. The total amount of function evaluations to obtain the *same* quality of solution seems to be the same for both recombination methods. We will investigate several recombination methods for continuous functions in a forthcoming paper.

The results of this section can be summarized as follows: *Recombination is an effective search operator in large populations with binary genes. For continuous functions recombination has linear order of convergence. But the optimum is only reached by an infinite population. Therefore recombination should not be used as the only search operator for continuous fitness functions.*

5 Analysis of strong selection and mutation

The mutation operator with a small number of parents is well understood. In principle it is just a problem of statistics - doing N trials in parallel instead of in a sequence. But selection converts the problem to a nonstandard statistical problem.

In this section we will analyse the behavior of mutation if the best individual only is used as parent of the next generation. This can be done in an elitist or non-elitist way. Elitist means that the former parent stays alive if all offspring are worse than the parent. In evolution strategies these two strategies are called the *plus strategy "+"* (elitist) and the *comma strategy ","* (Bäck & Schwefel, 1993). The simplest strategy is the $(1+1)-$strategy. It uses one parent and one offspring. The fitter of the two survives.

In (Mühlenbein, 1991; Mühlenbein, 1992), we computed the probability of a successful mutation for a single individual. From this analysis the optimal mutation rate was obtained. The optimal mutation rate maximizes the probability of a success. We just state the most important results.

Theorem 6 *For the ONEMAX function of size n the optimal mutation rate m is proportional to the size of the problem.*

$$m = \frac{1}{n}$$

This important result has been independently discovered several times. The implications of this result to biology and to evolutionary algorithms have been first investigated by (Bremermann, Rogson & Salaff, 1966).

The performance of recombination was measured by GEN_e, the number of generations until equilibrium. This measure cannot be used for mutation because the population will never converge to a unique genotype. Therefore we will use GEN_{opt} as performance measure for the mutation operator. It is defined as the average number of generations till the optimum has been found for the first time. For a population with two individuals (one parent and one offspring) GEN_{opt} has been computed by a Markov chain analysis (Mühlenbein, 1991; Mühlenbein, 1992). In this case GEN_{opt} is equal to FE_{opt}, the number of trials to reach the optimum.

Theorem 7 *Let p_0 be the probability of the advantageous allele in the initial string. Then the $(1+1)-$ strategy needs on the average the following number of trials FE_{opt}*

$$FE_{opt} = e \cdot n \sum_{j=1}^{(1-p_0)n} \frac{1}{j} \tag{21}$$

to reach the optimum of the ONEMAX function of size n. The mutation rate is set to $m = 1/n$.

Proof *We only sketch the proof. Let the given string have one incorrect bit left. Then the probability of switching this bit is given by*

$$s_1 = m \cdot (1-m)^{n-1} \approx e^{-1} \cdot m \tag{22}$$

The number of trials to obtain the optimum is given by $e \cdot 1/m$. Similarly if two bits are incorrect, then the number of trials needed to get one bit correct is given by $e/2 \cdot 1/m$.

The total number is obtained by summation.

Equation 21 can be approximated by

$$FE_{opt} = e \cdot n \cdot (ln\,((1-p_0)n) + \gamma) \tag{23}$$

We have confirmed the formula by intensive simulations (Mühlenbein, 1991). Recently Bäck (1993) has shown that FE_{opt} can be only marginally reduced if a theoretically optimal *variable* mutation rate is used. This variable rate depends on the number of bits yet to be corrected. This result has been predicted in (Mühlenbein, 1992). Mutation spends most of the time in adjusting the very last bits. But in this region the optimal mutation rate is $m = 1/n$.

Theorem 7 cannot easily be extended to a larger number of offspring. Therefore we will only qualitatively discuss this problem by simulations. Some results are displayed in table 3. One clearly observes the law of diminishing returns. Increasing the popsize N reduces GEN_{opt} less and less. Mutation is most efficient with a small number of offspring.

The non-elitist $(1, N)$ strategy has an interesting behavior. For very small N the strategy is almost a random walk. It requires a huge number of generations to randomly hit the optimum. For large N the comma strategy becomes similar to the plus strategy. There will always be an offspring of higher fitness than the parent. This shows that for the $(1, N)-$strategy there exists a unique $N > 1$ where the population needs the minimum number of function evaluations until convergence. For the $(1 + N)-$strategy this number is $N = 1$.

N	2	4	8	16	32	64	128	256	512	1024
(1+N)	750	400	200	115	71	48	37	30	26	23
(1,N)	-	-	241	112	69	50	38	30	26	23
S(1+N)	2.0	3.75	7.5	13.4	21	31	41	50	58	65

Table 3: GEN_{opt} and speedup S for mutation from one parent (ONEMAX(128))

The speedup S shows how much faster the solution is obtained with a larger number of offspring. It is defined as $GEN_{opt}(1)/GEN_{opt}(N)$. The speedup is almost linear for small N and seems to slow down to a logarithmic function. This indicates that mutation is not an efficient search in a large population. We will show in the next section that smaller selection intensities give still worse results.

The BGA mutation scheme for continuous functions has been analyzed in (Mühlenbein & Schlierkamp-Voosen, 1993b). It was shown that the $(1 + 1)-$strategy with the BGA mutation scheme has linear order of convergence. The same order of convergence was proven in the previous sections for discrete recombination.

6 Small truncation selection and mutation

First we will compare the severe selection of the previous section (only one individual is selected as parent) with constant truncation selection. In figure 4 the relation between

GEN_{opt}, FE_{opt}, and the popsize N is displayed for these two selection methods. The selection thresholds are $T = 0.5$ and the smallest one possible, $T = 1/N$. In large populations the strong selection outperforms the fixed selection scheme by far. These results can easily be explained. The mutation operator will change one bit on the average. The probability of a success becomes less the nearer the population comes to the optimum. Therefore the best strategy is to take just the best individual as parent of the next generation.

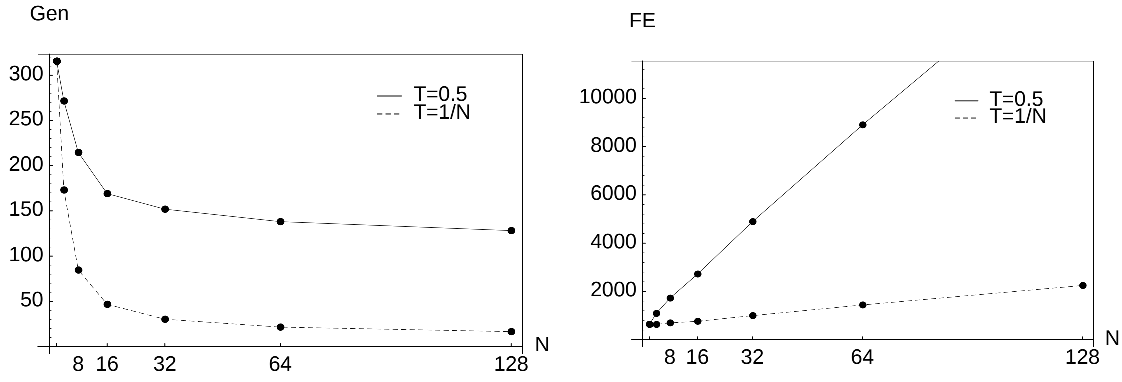

Figure 4: GEN_{opt} and function evaluations (FE) for various N and different T (ONEMAX(64))

From GEN_{opt} the expected number of trials needed to find the optimum can be computed

$$FE_{opt} = N \cdot GEN_{opt}$$

For both selection methods, FE_{opt} increases linearly with N for large N. The increase is much smaller for strong selection. The smallest number of function evaluations are obtained for $N \leq 4$.

We now turn to the theoretical analysis of mutation and truncation selection in a large population. The analysis depends on an extension of the response to selection equation.

Theorem 8 *Let s_t be the probability of a mutation success, imp the average improvement of a successful mutation. Let f_t be the probability that the offspring is worse than the parent, red the average reduction of the fitness. Then the response to selection for small mutations in large populations is given by*

$$R(t) = S(t) + s_t \cdot imp - f_t \cdot red \tag{24}$$

$S(t)$ is the average fitness of the selected parents.

Proof *Let $M_s(t)$ be the average of the selected parents. Then*

$$M(t+1) = s_t(M_s(t) + imp) + f_t(M_s(t) - red) + (1 - s_t - f_t)M_s(t)$$

Subtracting $M(t)$ from both sides of the equation we obtain the theorem.

The response to selection equation for mutation contains no *heritability*. Instead there is an offset, defined by the difference of the probabilities of getting better or worse. The importance of s_t and f_t has been independently discovered by Schaffer & Eshelman, (1991). They did not use the difference of the probabilities, but the quotient which they called the *safety factor.*

$$F = \frac{s_t}{f_t}$$

In order to obtain an empirical law we have to estimate s_t and f_t. This can be done by using the results of (Mühlenbein, 1991). The estimation requires the average number i of wrong bits of the parent strings as input. But i can be easily transformed into a variable depending on the state of the population at generation t. This variable is the marginal probability $p(t)$ that there is the advantageous allele at a locus. $p(t)$ was already used in the previous theorems. i and $p(t)$ are connected by

$$i \approx n \cdot (1 - p(t)) = n - M(t) \tag{25}$$

The following empirical law will be derived by a mixture of statistical analysis and simulation results.

Empirical Law 1 *For a truncation threshold of $T = 0.5$, a mutation rate of $m = 1/n$, and $n \gg 1$ the response to selection of a large population changing by mutation is approximately*

$$R(t) = 1 + (1 - p(t))e^{-p(t)} - p(t)e^{-(1-p(t))} \tag{26}$$

Proof *Let each parent have i bits wrong, let s_i be the probability of a success by mutation, f_i be the probability of a defect mutation. s_i is approximately given by the product of changing at least one of the wrong bits while not changing an correct bit (Mühlenbein, 1991). Therefore*

$$s_i = (1 - m)^{n-i}(1 - (1 - m)^i)$$

Similarly

$$f_i = (1 - m)^i(1 - (1 - m)^{n-i})$$

We now turn to the population at generation t. From equation 25 and $1 - (1 - m)^i \approx i \cdot m$ we obtain

$$s_t = (1 - p(t))(1 - \frac{1}{n})^{np(t)}$$

$$f_t = p(t)(1 - \frac{1}{n})^{n(1-p(t))}$$

Because $(1 - \frac{1}{n})^n \approx e^{-1}$ we get

$$s_t = (1 - p(t))\, e^{-p(t)}$$

$$f_t = p(t)e^{-(1-p(t))}$$

We are left with the problem of estimating imp and red. In a first approximation we set both to 1 because a mutation rate of $m = 1/n$ changes one bit on the average. We have not been able to estimate S(t) analytically. Simulations show that for $T = 0.5$ S(t) *decreases from about 1.15 at the beginning to about 0.9 at GEN_{opt}. Therefore $S(t) = 1$ is a reasonable approximation. This completes the proof.*

Equation 26 defines a difference equation for $p(t+1)$. We did not succeed in solving it analytically. We have found that the following linear approximation gives almost the same results

Empirical Law 2 *Under the assumptions of empirical law 1 the response to selection can be approximated by*

$$R(t) = 2 - 2p(t) \tag{27}$$

The number of generations until $p(t) = 1 - \epsilon$ is reached is given by

$$GEN_{1-\epsilon} \approx \frac{n}{2} \cdot ln \frac{1 - p_0}{\epsilon} \tag{28}$$

Proof: *The proof is identical to the proof of theorem 5.*

In figure 5 the development of the mean fitness is shown. The simulations have been done with two popsizes ($N = 1024$ and 64) and two mutation rates ($m = 1/n$ and $4/n$). The agreement between theory and simulation is fairly good. The evolution of the mean fitness of the large population and the small population is more or less the same, if the same mutation rate and the same truncation threshold is used. This result demonstrates that a large population is computationally inefficient for mutation.

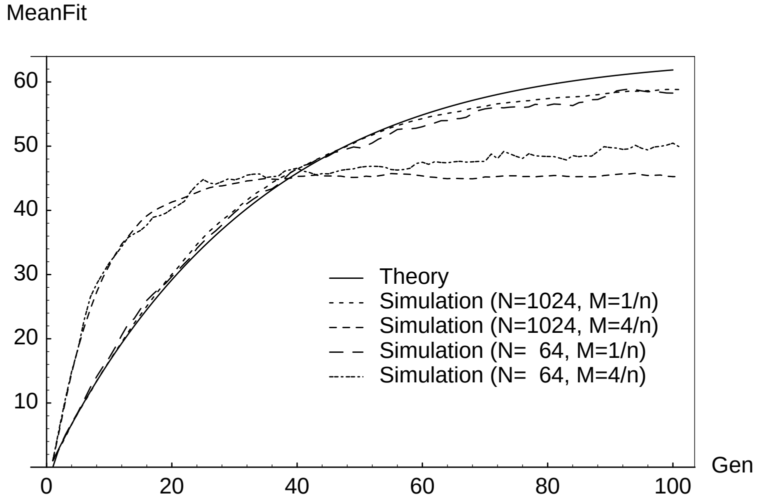

Figure 5: Mean fitness for small and large N and mutation rates of $1/n$ and $4/n$ ($p_0 = 1/64$). The population size has no influence on the increase of the meanfitness. The mean fitness of the population with mutation rate $4/n$ increases faster, but the increase stops at generation 40.

A large mutation rate has an interesting effect. The mean fitness increases faster at the beginning, but the population never finds the optimum. This observation again suggests using a variable mutation rate. But we have already mentioned that the increase in performance by using a variable mutation rate is rather small. Mutation spends most of its time in getting the last bits correct. But in this region a mutation rate of $m = 1/n$ is optimal.

The major results of the last two sections can be summarized as follows: *Mutation in large populations is not effective. It is more efficient with very strong selection. The response to selection becomes very small when the population is approaching the optimum. The efficiency of the mutation operator critically depends on the mutation rate.*

7 Summary of the major results

Let n denote the number of genes, N the size of the population. Any finite population of size N will converge to a single genotype, even if selection is not applied. This effect is called *genetic drift.* The expected number of generations until convergence GEN_e is surprisingly low.

$$E(GEN_e) \approx 1.4 \cdot N \cdot (0.5ln(n) + 1)^{1.1} \tag{29}$$

The above equation is valid for recombination without selection and mutation. We now turn to truncation selection. If the size N of the population is larger than the *critical popsize* N^*, the minimum popsize to converge to the optimum with high probability, then we have for the expected number of generations until convergence

$$GEN_e \approx \left(\frac{\pi}{2} - arcsin(2p_0 - 1)\right) \cdot \frac{\sqrt{n}}{I} \tag{30}$$

In the above equation the mutation rate is set to 0. Note that GEN_e is independent of N. The estimation of the critical popsize is very difficult. We conjecture

$$N^* \approx \sqrt{n} \cdot ln(n) \cdot f_1(p_0) \cdot f_2(I) \tag{31}$$

Proportional selection as used by the simple GA (Goldberg, 1989) selects too weakly when the variance of the population becomes small. The expected number of generations $GEN_{1-1/n}$ until the favorable allele is distributed in the population with probability of $1 - 1/n$ is given by

$$GEN_{1-1/n} \approx n \cdot ln(n \cdot (1 - p_0)) \tag{32}$$

This equation is valid for a mutation rate of 0. The number of generations is much larger than with truncation selection. The analysis of recombination in small populations is difficult. We have shown some results in phase diagrams relating the posize and GEN_e. The phase diagrams can be divided into two areas. The border is given by the critical popsize N^*.
We now turn to populations using only mutation. Mutation is a random search operator especially efficient in small populations. The most important result concerns the mutation rate. The mutation rate is defined as the probability of mutating a gene.

Rule of thumb: *The mutation rate $m = 1/n$ where n is the size of the chromosome is almost optimal .*

For the above mutation rate the expected number of generations GEN_{opt} until the optimum is found has been computed for the $(1+1)-$strategy (one parent, one offspring; the better of the two survives).

$$GEN_{opt} \approx e \cdot n \cdot (ln(n \cdot (1-p_0)) + \gamma) \tag{33}$$

Mutation in a large population is inefficient. The asymptotic scaling of GEN_{opt} is independent of the popsize N. It stays at $O(n \cdot ln(n))$. For very large popsizes GEN_{opt} is given by

$$GEN_{1-1/n} \approx \frac{n}{2} \cdot ln(n \cdot (1-p_0)) \tag{34}$$

The above equation is valid for a large population and a truncation selection threshold of $T = 0.5$. Note that the above value is about half the value of proportional selection.

The above theorems show that for binary representations populations using either recombination or mutation are able to locate the optimum. If $p_0 = 0.5$ i.e. half of the bits are correct in the initial population, the asymptotic order of the number of trials needed (FE_{opt}), seems to be the same, namely $O(n \cdot ln(n))$. For recombination this number is obtained by multiplying GEN with the critical popsize N^*. Therefore the question which of the two operators is more efficient is difficult to answer. The comparison needs an exact expression for N^*, which we have not yet obtained. But we can easily make a qualitative comparison. The major difference between mutation and recombination is their dependence on p_0, the percentage of the desired allele in the initial population.

Let us take $p_0 = 1 - 1/n$ as example. Here one bit is wrong on the average. Mutation will need about $O(n)$ trials to change the incorrect bit. Uniform crossover of two strings, each with one bit wrong, will generate the optimum string with probability $1/4$, independent of the size of the problem. Therefore the critical popsize N^* is also independent of n. Thus recombination is much more efficient than mutation. But the determination of the exact N^* is also difficult in this simple case. It will need on the average 4 trials to generate the optimum. But the probability that a popsize of 4 will not generate the optimum is $0.75^4 = 0.31$. It needs 16 trials in order to obtain the optimum with 99% probability.

If we take $p_0 = 1/n$ the situation is reversed. Now mutation is much more efficient than recombination which needs a huge popsize in order to locate the optimum. This behavior can also be found in individual runs. By comparing figures 1 and 5 one observes that the increase of the average fitness of a population using mutation is better than that of a population using recombination when far away from the optimum. Recombination has too few building blocks to generate better offspring. But recombination is more effective than mutation nearby the optimum. Here the likelihood of success of mutation is much lower.

A more detailed comparison between mutation and recombination, also by means of a competition between populations can be found in (Mühlenbein & Schlierkamp-Voosen, 1993a). The question now arises how to combine mutation and recombination so that the resulting algorithm is more effective than an algorithm using a single genetic operator. We see two approaches at least.

In the first approach, an optimal mutation rate is constantly applied. The variance of the population remains high enough for recombination to be effective. This method is normally used by the BGA. The second approach is mainly based on recombination. If the variance is below a certain threshold the population is thoroughly changed by applying mutation vigorously. This event gives recombination the chance for further improvements. This approach is used by Eshelman (1991) in the CHC algorithm. The success of this procedure depends on the right amount of change. If too much is changed then this would be just a new start of the algorithm. If the changes are too small then the population will stay in equilibrium.

Most of our analytical results have been derived under the assumption of additive genetic effects. This theory explains the behavior of the most important evolutionary forces. It plays a similar role for the BGA as the "ideal gas" theory for thermodynamics. There exists no *ideal gas* in reality, but the ideal gas theory gives much insight into the overall behavior of gases. In order to understand evolutionary algorithms in more complex fitness landscapes, we have to extend the theory by using more advanced statistical methods. This is the topic of the next section.

8 Statistics and genetics

In this section we will present two methods for estimating the *heritability*. The first one will use the concept of *regression of offspring to parent* and the second one the concept of *genetic variance*. Both methods have been of great importance in the development of statistics and population genetics. Therefore we will first give a short historical survey.

Genetics represents one of the most satisfying applications of statistical methods. Modern statistics starts with Galton and Pearson who found at the end of the last century a striking empirical regularity. On the average a son is halfway between his father and the overall average height for sons. They used data from about 1000 families. In order to see this regularity Galton and Pearson invented the scatter diagram, regression and correlation (see Freedman et al., 1991).

Independently Mendel found some other striking empirical regularities like the reappearance of a recessive trait in one-fourth of the second generation hybrids. He made up a chance model involving what are now called *genes* to explain his rules. He conjectured these genes by pure reasoning - he never saw any.

At first sight, the Galton-Pearson results look very different from Mendel's, and it is hard to see how they can be explained by the same biological mechanism. Indeed Pearson wrote an article in 1904 claiming that his results cannot be derived by Mendel's laws. About 1920 Fisher, Wright and Haldane more or less simultaneously recognized the need to recast the Darwinian theory as described by Galton and Pearson in Mendelian terms. They succeeded in this task, but unfortunately much of the original work is abstruse and very difficult to follow. The difficulty lies in the exact definition of *genetic variance* and its connection to *heritability*. We will in this section adapt the classical methods to haploid chromosomes. Furthermore we will precisely define the concepts.

The first theorem connects the realized heritability $b_t = R(t)/S(t)$ with the regression coefficient between *midparent* and offspring. Let f_i, f_j be the phenotypic values of parents i and j, then

$$\bar{f_{i,j}} = \frac{f_i + f_j}{2}$$

is called the midparent value. Let the stochastic variable $\bar{F}$ denote the midparent value.

Theorem 9 *Let $F(t) = (f_1, \ldots, f_N)$ be the population at generation t, where f_i denotes the phenotypic value of individual i. Assume that an offspring generation $O(t)$ is created by random mating, without selection. If the regression equation*

$$o_{ij}(t) = a(t) + b_{\bar{F}O}(t) \cdot \frac{f_i + f_j}{2} + \epsilon_{ij} \tag{35}$$

with

$$E(\epsilon_{ij}) = 0$$

is valid, where o_{ij} is the fitness value of the offspring of i and j, then

$$b_{\bar{F}O}(t) \approx b_t \tag{36}$$

Proof *From the regression equation we obtain for the expected averages*

$$E(O(t)) = a(t) + b_{\bar{F}O}(t)M(t)$$

Because the offspring generation is created by random mating without selection, the expected average fitness remains constant

$$E(O(t)) = M(t)$$

Let us now select a subset as parents. The parents will be randomly mated, producing the offspring generation. If the subset is large enough, we may use the regression equation and obtain for the averages

$$M(t+1) = a(t) + b_{\bar{F}O}(t) \cdot M_s(t)$$

Here $M(t+1)$ is the average fitness of the offspring generation produced by the selected parents. Subtracting the above equations we obtain

$$M(t+1) - M(t) = b_{\bar{F}O}(t) \cdot (M_s(t) - M(t))$$

This proves $b_{\bar{F}O}(t) = b_t$.

The importance of regression for estimating the heritability was discovered by Galton and Pearson. They computed the regression coefficient rather intuitively by scatter diagrams of midparent and offspring (see Freedman et al., 1991). The problem of computing a good regression coefficient is solved by the theorem of Gauss-Markov. We just cite the theorem. The proof can be found in any textbook on statistics (Rao, 1973).

Theorem 10 *A good estimate for the regression coefficient of midparent and offspring is given by*

$$b_{\bar{F}O}(t) = \frac{cov(O(t), \bar{F}(t))}{var(\bar{F}(t))} \tag{37}$$

The covariance of O and $\bar{F}$ is defined by

$$cov(O(t), \bar{F}(t)) = \frac{1}{N} \sum_{i,j} (o_{i,j} - av(O(t))) \cdot (\bar{f}_{i,j} - av(\bar{F}(t)))$$

av denotes the average and var the variance. Closely related to the regression coefficient is the correlation coefficient $cor(\bar{F}, O)$. It is given by

$$cor(\bar{F}(t), O(t)) = b_{\bar{F}O}(t) \cdot (\frac{var(\bar{F}(t))}{var(O(t))})^{1/2}$$

The above theorem enables us to estimate the heritability by a second method. It works as follows. For a large sample population F the offspring have to be created by random mating. Then the regression coefficient $b_{\bar{F}O}$ can be computed by equation 37. This procedure is more robust than dividing $R(t)$ by $S(t)$. First, it works also in the case of small selection intensity. Second, the trustworthiness of the computation can be estimated by statistical techniques.

By the above method an average value for the heritability is computed. The average is taken over the whole domain. For the breeder genetic algorithm we decided to proceed slightly differently. The regression coefficient is only computed for the *selected parents* and their offspring. This local approximation makes it possible to compute regression coefficients which depend on the given population and the local fitness landscape.

The next theorem shows the connection between *midparent* and *parent* regression.

Theorem 11 *Midparent and parent regression are connected by*

$$b_{FO}(t) = 0.5 \cdot b_{\bar{F}O}(t) \quad cor(F(t), O(t)) = \sqrt{\frac{1}{2}} cor(\bar{F}(t), O(t)) \tag{38}$$

Proof *We have*

$$cov(O(t), \bar{F}(t)) = cov(O(t), F(t))$$
$$var(\bar{F}(t)) = 0.5 \cdot var(F(t))$$

From (37) the theorem is obtained.

We now describe a method for estimating the covariance. This method connects a microscopic genetic chance model and the macroscopic phenotypic covariance. It is restricted to discrete genes. In this paper we only give the necessary definitions and the fundamental theorem. The interested reader is refered to Asoh and Mühlenbein (1994a) where the proof can be found. A detailed computation is given for a diploid chromosome with two genes in (Crow & Kimura, 1970).

Let a haploid chromosome with n binary genes x_i be given, $f(\mathbf{x})$ its fitness. Let the genetic chance model be defined by *uniform crossover*. This model can be considered as Mendel's chance model restricted to haploid chromosomes. We will decompose the fitness value $f(\mathbf{x})$ recursively into an additive part and interaction parts. Let $p(\mathbf{x})$ denote the probability of $\mathbf{x}$, $p(\mathbf{x}|x_i)$ the conditional probability of $\mathbf{x}$ given x_i. First we extract the average.

$$f(\mathbf{x}) = av(f) + r_0(\mathbf{x}) \tag{39}$$

Then we extract the first order (additive) part from the residual $r_0(\mathbf{x})$.

$$r_0(\mathbf{x}) = \sum_{i=1}^{n} f_{(i)}(x_i) + r_1(\mathbf{x}) \tag{40}$$

where $f_{(i)}(x_i)$ are given by

$$f_{(i)}(x_i) = \sum_{\mathbf{x}|x_i} p(\mathbf{x}|x_i) r_0(\mathbf{x}) = \sum_{\mathbf{x}|x_i} p(\mathbf{x}|x_i) f(\mathbf{x}) - av(f)$$

Here $\sum_{\mathbf{x}|x_i}$ means that the i-th locus is fixed to the value x_i. The $f_{(i)}(x_i)$ minimize the quadratic error $\sum_{\mathbf{x}} p(\mathbf{x}) r_1(\mathbf{x})^2$.
If $r_1(\mathbf{x}) \not\equiv 0$, we can proceed further to extract the second order terms from $r_1(\mathbf{x})$:

$$r_1(\mathbf{x}) = \sum_{\substack{(i,j)\\ i<j}} f_{(i,j)}(x_i, x_j) + r_2(\mathbf{x}) \tag{41}$$

where

$$\begin{aligned} f_{(i,j)}(x_i, x_j) &= \sum_{\mathbf{x}|x_i,x_j} p(\mathbf{x}|x_i, x_j)\ r_1(\mathbf{x}) \\ &= \sum_{\mathbf{x}|x_i,x_j} p(\mathbf{x}|x_i, x_j)\ f(\mathbf{x}) - f_{(i)}(x_i) - f_{(j)}(x_j) \end{aligned}$$

If we have n loci, we can iterate this procedure $n-1$ times recursively and finally we get the decomposition of f as

$$\begin{aligned} f(\mathbf{x}) &= \bar{f} + \sum_{i} f_{(i)}(x_i) + \sum_{(i,j)} f_{(i,j)}(x_i, x_j) + \cdots \\ &+ \sum_{\substack{(i_1,\ldots,i_{n-1})\\ i_1<\ldots<i_{n-1}}} f_{(i_1,\ldots,i_{n-1})}(x_{i_1}, \ldots, x_{i_{n-1}}) + r_{n-1}(\mathbf{x}) \end{aligned}$$

Let V_k for $k = 1$ to $n-1$ be defined as

$$V_k = \sum_{\substack{(i_i,\ldots,i_k)\\ i_1<\ldots<i_k}} \sum_{x_{i_1},\ldots,x_{i_k}} p(x_{i_1}, \ldots, x_{i_k}) f_{(i_i,\ldots,i_k)}(x_{i_i}, \ldots, x_{i_k})^2, \tag{42}$$

and

$$V_n = \sum_{\mathbf{x}} p(\mathbf{x}) r_{n-1}(\mathbf{x})^2 \tag{43}$$

We are now able to formulate the fundamental theorem.

Theorem 12 *Let the population be in linkage equilibrium i.e.*

$$p(\mathbf{x}) = \prod_{i=1}^{n} p_i(x_i) \tag{44}$$

Then the variance of the population is given by

$$var(F) = V_1 + V_2 + \cdots + V_{n-1} + V_n \tag{45}$$

The covariance of midparent and offspring can be computed from

$$cov(\bar{F}, o) = \frac{1}{2}V_1 + \frac{1}{4}V_2 + \cdots + \frac{1}{2^n}V_n = \sum_{k=1}^{n} \frac{1}{2^k}V_k \tag{46}$$

From theorems 10 and 12 we obtain

Corollary 1 *If the fitness function is additive that is,* $f(\mathbf{x}) = \sum_i f_i(x_i)$, *then*

$$cor(\bar{F}, O) = \sqrt{1/2} \qquad b_{\bar{F}O} = 1 \tag{47}$$

The above theorem plays an important role in the science of breeding. Breeders conjecture that the *additive genetic variance* V_1 is the most important factor of the heritability. The higher order interactions contribute much less to the heritability. Therefore they can be neglected. We will test this conjecture in a forthcoming paper.

Numerically, decomposing the variance is computationally far too expensive to be of use for the breeder genetic algorithm. But the regression technique is very simple to implement. We will show in the next section that the regression technique can be used to control and guide the breeder genetic algorithm.

9 Numerical applications of the theory

From statistics and population genetics it is known that the regression coefficient should be a reliable estimate for heritability in the case of continuous fitness functions and large populations. Therefore as a first example we take the minimization of the hypersphere. The BGA for continuous functions has been described in (Mühlenbein & Schlierkamp-Voosen, 1993b). It uses a floating point representation. In figure 6 scatter diagrams of midparent and offspring at generation 1 and 30 are shown. In this example only discrete recombination is used, no mutation. It is easily seen that the whole population is moving towards the global minimum, which is 0 in this example. The regression coefficient is almost exactly one in both diagrams as predicted by the theory.

In figure 7 the numerical values of the two different estimates for the heritability are shown ($R(t)/S(t)$ and the regression coefficient). Both estimates oscillate around 1 as predicted. The correlation coefficient is about 0.5. This is less than the maximum value possible, which is $\sqrt{0.5}$. The reason for this difference is the selection. The selection reduces the variance of the parents and therefore the correlation coefficient.

We just report the results for a simulation run without selection. In this case the $R(t)/S(t)$ estimator cannot be used because S(t) is about 0. The regression coefficient can be computed as usual and remains 1. Furthermore the correlation coefficient is about $\sqrt{0.5}$ as predicted by the theory.

The above results are not restricted to simple unimodal functions. As the next example we take the highly multimodal function which is known as Schwefel's function $F7$.

$$F_7 = \sum_{1}^{n} -x_i \; sin\left(\sqrt{|x_i|}\right) \qquad -500 \leq x_i \leq 500 \tag{48}$$

The theory predicts that the multimodality of this function can be considered more or less as noise for the BGA. It should have no major influence on the regression coefficient.

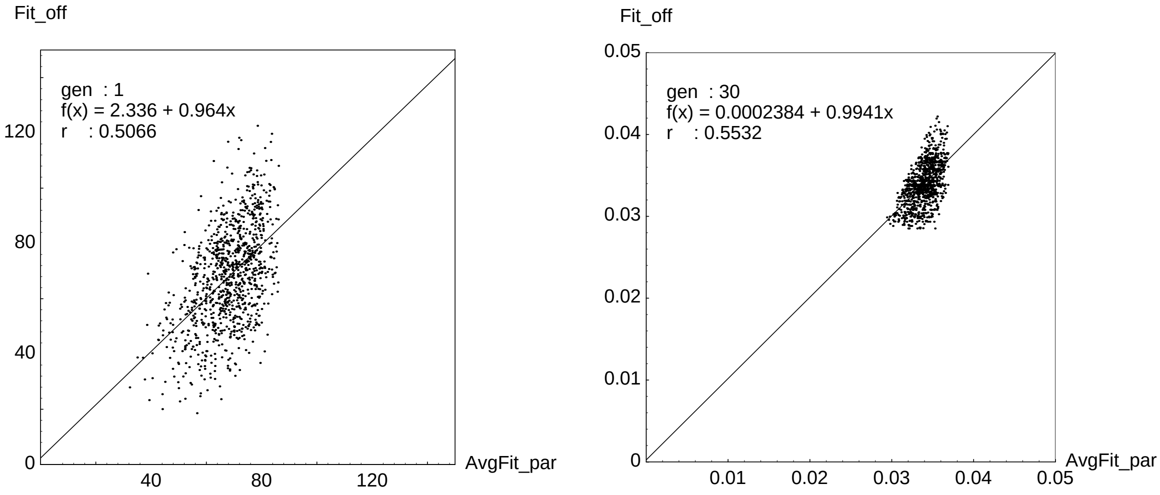

Figure 6: Scatter diagrams for generations 1 and 30 for the hypersphere. Only discrete recombination is used (N=1024,T=0.5).

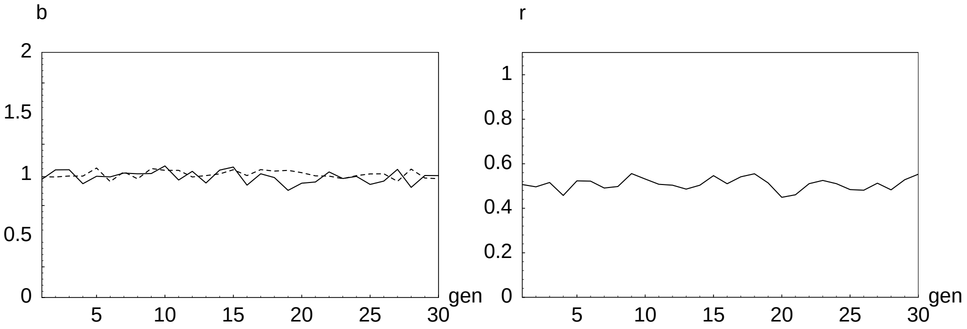

Figure 7: Heritability estimates (regression coefficient solid line, $R(t)/S(t)$ dashed line) and correlation coefficient r for the hypersphere (N=1024,T=0.5).

Indeed, with random mating, the regression coefficient is 1 and the correlation coefficient between midparent and parent is about $\sqrt{0.5}$, just as for the hypersphere. Figure 8 shows a real BGA simulation run with selection, recombination *and* mutation. One clearly observes that the search is first driven by recombination, then by mutation. From generation 17 on, the regression coefficient substantially differs from the ratio estimator $R(t)/S(t)$. Now the search is mainly driven by the random operator mutation. The BGA mutation scheme is described in (Mühlenbein & Schlierkamp-Voosen, 1993).

Next we turn to binary functions. We take as examples

- ONEMAX(n)
- PLATEAU(20,3)
- DECEP(10,3)

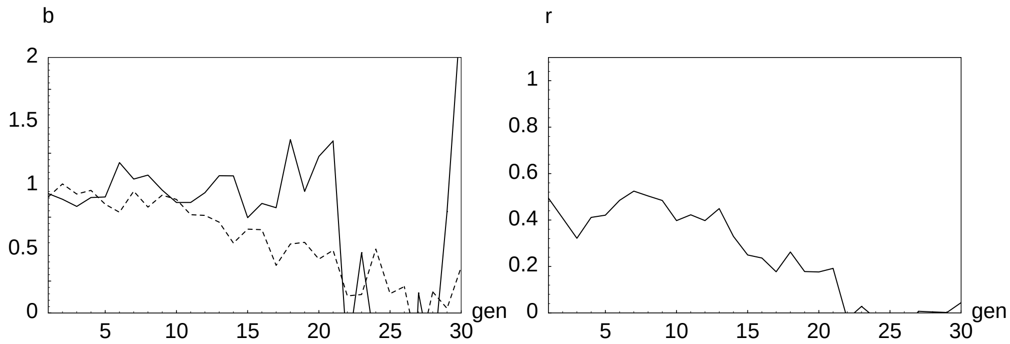

Figure 8: Heritability estimates b with mutation and recombination ($N = 256$). The correlation coefficient r drops to zero. The regression coefficient (solid line) and the ratio estimator (dashed line) are almost equal at the beginning. Then the ratio $R(t)/S(t)$ goes to zero whereas the regression coefficient remains high till generation 22.

$PLATEAU(20,3)$ has a string length n of 60. An increase in fitness is allocated only if three consecutive bits at loci 1,3,6,.. are 1's. In each case, the fitness is increased by 3. $DECEP(10,3)$ is the deceptive function defined by Goldberg (Mühlenbein, 1991).

In figure 9 the results of a BGA run are shown for $ONEMAX(64)$ with a truncation threshold of $T = 0.5$ and uniform crossover, but without mutation. The two heritability estimates coincide fairly well. They are about 1, as predicted. The correlation coefficient is about 0.5 till generation 14. This is less than the correlation coefficient without selection, which is $\sqrt{0.5}$. At the end of the run the correlation coefficient increases. This behavior indicates that the genotypes of the selected parents are becoming very similar. Therefore the offspring are very similar to both parents.

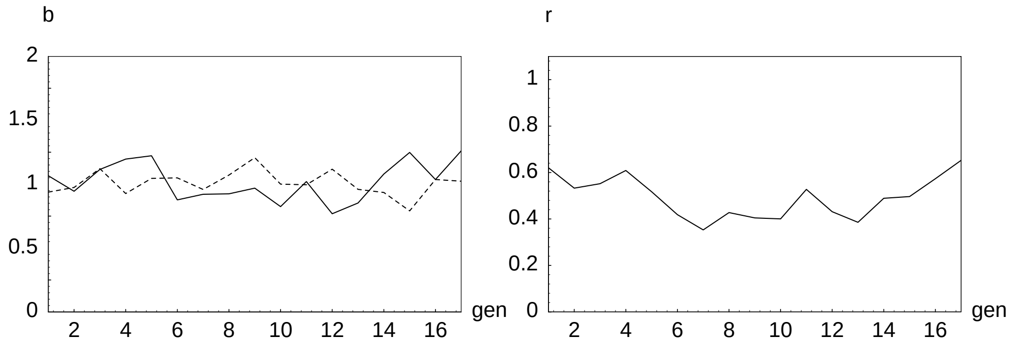

Figure 9: Heritability b estimates (regression coefficient solid line, $R(t)/S(t)$ dashed line) and correlation coefficient r with recombination only for ONEMAX(64) ($N = 128, T = 0.5$)

Our next example is the PLATEAU function. We will discuss PLATEAU(20,3) and PLATEAU(20,5). PLATEAU(20,5) has a plateau of size 5, therefore it is more difficult to optimize. Without selection the regression coefficients for the two functions are about 0.7 and 0.4, the correlation coefficients are about 0.5 and 0.3. In figure 10 we have used a truncation threshold of $T = 0.5$. For both functions the regression coefficients are substantially higher than without selection. This indicates that selection is very effective for this fitness function. But note that the realized heritability $R(t)/S(t)$ is considerably smaller than the regression coefficient. For PLATEAU(20,5) it substantially increases during the run.

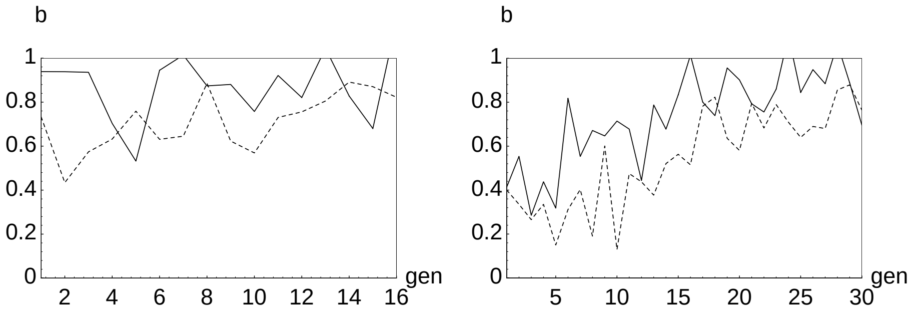

Figure 10: Heritability b estimates (regression coefficient solid line) for PLATEAU(20,3) and (20,5)

The last example is the deceptive function DECEP(10,3). This function is called deceptive, because the search is guided into the local optimum $(0,0,0)$. The global optimum is at $(1,1,1)$. Without selection, the regression coefficient is about 0.5 and the correlation coefficient about 0.35. This is shown in figure 11.

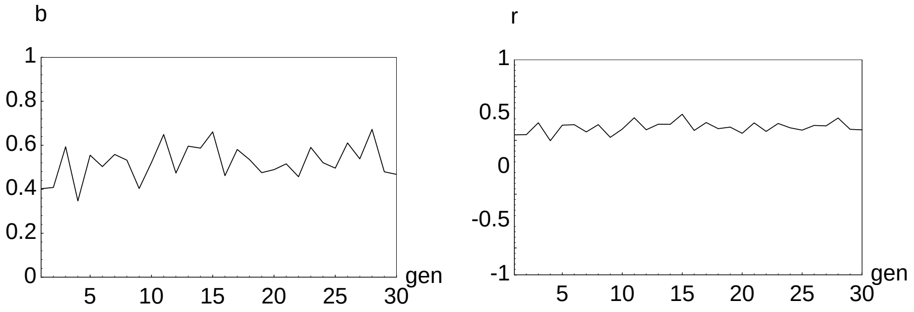

Figure 11: Heritability b and correlation r estimate with recombination for DECEP(10,3), no selection ($N = 256$)

The behavior radically changes with selection. If selection is applied, both the regression coefficient and the ratio estimator become erratic. Half of the time they are negative. This shows selection with this fitness function works against crossover and vice versa.

For binary functions the heritability can also be estimated by decomposing the genetic variance. We have already used this method for the ONEMAX function. But the numerical implementation for the general case is prohibitive. The method of decomposing the variance will numerically be useful if the first term, the additive genetic variance V_1 is sufficient for estimating the heritability. We must postpone this investigation.

To summarize this section:*The theory presented is especially applicable for continuous functions. For many continuous fitness functions the regression coefficient will be* 1, *the maximum possible. For binary functions the regression coefficient and the realized heritability give useful information about the complexity of the fitness landscape and how to guide the search of the breeder genetic algorithm.*

10 Conclusion

Efficient evolutionary algorithms for optimization should be based on the science of breeding animals rather than on natural selection. In this paper we have adapted some of the scientific methods used by breeders for our Breeder Genetic Algorithm BGA. Some of the results, already known in the science of breeding, have been extended or made more precise. Some results presented in this paper, e.g. the optimal mutation rate, are unique to evolutionary algorithms. The breeder cannot influence mutation or recombination. For breeding of animals recombination plays the most important part.

The BGA tries to solve the problem of how to scientifically breed a virtual population. For genetic representations similar to the ones used in population genetics a predictive theory was developed. But these representations are not the only ones imaginable for optimization problems. Our research in the future will concentrate on one of the difficult problems remaining - how to find a good genetic representation for a given application. Some representations may also profit from new genetic operators which do not have a counterpart in nature. An example is the exchange of subtrees which is used in genetic programming. This operator works on chromosomes of varying length. We will try to investigate this operator using the framework presented in this paper.

Acknowledgement: Part of this research has been funded by the REAL WORLD COMPUTING programme under the project SIFOGA. The authors thank H. Asoh for his contributions concerning genetic drift and the variance decomposition.

References

Asoh, H., & Mühlenbein, H. (1994a). Estimating the heritability by decomposing the genetic variance. Technical report. Sankt Augustin: GMD.

Asoh, H., & Mühlenbein, H. (1994b). On the mean convergence time of genetic populations without selection. Technical report. Sankt Augustin: GMD.

Bäck, Th. (1993). Optimal mutation rates in genetic search. In S. Forrest (Ed.), *5rd Int. Conf. on Genetic Algorithms*, (pp. 2-9), San Mateo: Morgan Kaufmann.

Bäck, Th. & Schwefel, H.-P. (1993). An overview of evolutionary algorithms for parameteroptimization. *Evolutionary Computation*, 1:1–24.

H.J. Bremermann, H.J., Rogson, J. & Salaff, S. (1966). Global properties of evolution processes. In H.H. Pattee (Ed.), *Natural Automata and Useful Simulations*, (pp. 3-42), Washington, DC: Spartan Books.

Bulmer, M.G. (1980). *"The Mathematical Theory of Quantitative Genetics"*. Oxford: Clarendon Press.

Crow, J.F. & Kimura, M. (1970). *An Introduction to Population Genetics Theory.* New York: Harper and Row.

Darwin Ch. (1859). *The Origins of Species by Means of Natural Selection.* London: Penguin Classics.

Eshelman L.J. (1991). The CHC Adaptive Search Algorithm: How to Have Safe Search when Engaging in Nontraditional Genetic Recombination. In G. Rawlins (Ed.), *Foundations of Genetic Algorithms*, (pp. 265-283), San Mateo: Morgan-Kaufman.

Freedman, D., Pisani,R., Purves, R., & Adhikkari, A. (1991). *Statistics second edition.* New York: W.W. Norton.

D.E. Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning.* Reading: Addison-Wesley.

Kimura,M. (1983). *The neutral theory of molecular evolution.* Cambridge: Cambridge University Press.

Mühlenbein,H., & Schlierkamp-Voosen,D. (1993a). Analysis of Selection, Mutation and Recombination in Genetic Algorithms. *Neural Network World*, 3:907–933.

Mühlenbein,H., & Schlierkamp-Voosen, D. (1993b). Predictive Models for the Breeder Genetic Algorithm: Continuous Parameter Optimization. *Evolutionary Computation*, 1:25–49.

Mühlenbein, H.(1991). Evolution in Time and Space - the Parallel Genetic Algorithm. In G. Rawlins, editor, *Foundations of Genetic Algorithms*, (pp. 316–337), San Mateo: Morgan-Kaufman.

Mühlenbein,H.(1992). How Genetic Algorithms Really Work: Mutation and Hill-climbing. In R. Männer and B. Manderick, editors, *Parallel Problem Solving from Nature*, (pp. 15-26), Amsterdam: North-Holland.

Rao, C.R.(1973). *Linear Statistical Inference and Its Application.* New York: Wiley.

Schaffer, J.D., & Eshelman, L.J.(1991). On crossover as an evolutionary viable strategy. In R. K. Belew & L. Booker (Eds.), *Procedings of the Fourth International Conference on Genetic Algorithms*, (pp. 61-68), San Mateo: Morgan Kaufmann.

Syswerda, G.(1989). Uniform crossover in genetic algorithms. H. Schaffer, (Ed.), *3rd Int. Conf. on Genetic Algorithms*, (pp. 2-9), San Mateo: Morgan Kaufmann.

Verrier, E., Colleau, J.J.,& Foulley, J.L.(1991). Methods for predicting response to selection in small populations under additive genetic models: a review. *Livestock Production Science*, 29:93-114.

Wright, S.(1932). The roles of mutation, inbreeding, crossbreeding and selection in evolution. *Proc. 6th Int. Congr. on Genetics*, (pp. 356–366).