Self–Adaptation in Genetic Algorithms

Thomas Bäck*

University of Dortmund
Department of Computer Science · Chair of Systems Analysis
P.O. Box 50 05 00 · D–4600 Dortmund 50 · Germany

Abstract

Within *Genetic Algorithms* (GAs) the mutation rate is mostly handled as a global, external parameter, which is constant over time or exogeneously changed over time.

In this paper a new approach is presented, which transfers a basic idea from *Evolution Strategies* (ESs) to GAs. Mutation rates are changed into endogeneous items which are adapting during the search process. First experimental results are presented, which indicate that environment–dependent *self–adaptation* of appropriate settings for the mutation rate is possible even for GAs.

Furthermore, the reduction of the number of external parameters of a GA is seen as a first step towards achieving a problem–dependent self–adaptation of the algorithm.

Introduction

Natural evolution has proven to be a powerful mechanism for emergence and improvement of the living beings on our planet by performing a randomized search in the space of possible DNA-sequences. Due to this knowledge about the qualities of natural evolution, some researchers tried to use the basic mechanisms of evolution as a basis of optimum-seeking techniques in case of vast search spaces.

In general, the algorithmic models of Darwinian evolution maintain a *population* of *individuals* (biological terminology has been adopted in this field). The population is able to adapt to a given (static or dynamically changing) environment by randomized processes of *selection*, *reproduction*, sexual *recombination*, and *mutation*. The environment provides a quality information (*fitness*) for the individuals, and the selection process favours the individuals of higher quality to survive ("survival of the fittest"). Even the reproduction process often favours structures of higher quality. Thus, during the evolution the average quality of the population increases, hopefully leading to an optimum solution.

Algorithms following this general approach have been summarized under the term *Evolutionary Algorithms* elsewhere, e.g. see Mühlenbein (1991) or Bäck and Hoffmeister (1991b). They include the *Genetic Algorithm* (GA) by Holland (1975) and the *Evolution Strategy* (ES) by Rechenberg (1973) and Schwefel (1981) as main representatives. At first glance they are mainly differing with respect to the structure of the individuals, but ESs benefit from the additional capability of learning on the level of strategy parameters by *self-adapting* them during the search (*second-level learning*). A detailed comparison of both algorithms was presented by Hoffmeister and Bäck (1991).

Within this paper a first effort towards incorporating the feature of self-adaptation into GAs by using adaptive mutation rates is presented. The general idea is to depart from global, fixed control mechanisms as used in GAs and instead to decentralize control by spreading it over the individuals. Additionally, control rules are no longer fixed within this distributed control approach, but control is subject to individual adaptation, thus facilitating a self–organizing behaviour of the population. Individually emerging behaviour instead of global control was identified by Langton (1989) to be one of the main characteristics of Artificial Life, hence the border (if any) between GAs and AL dissolves by means of self-adaptation.

Adaptive Mutation Rates

Much experimental work has been done in order to determine the best setting for the bit-mutation probability p_m of a Genetic Algorithm, but no clear answer

*baeck@ls11.informatik.uni-dortmund.de

to this question could be given. Some common settings are $p_m = 0.001$ (De Jong (1975)), $p_m = 0.01$ (Grefenstette (1986)), and $p_m \in [0.005, 0.01]$ (Schaffer, Caruna, Eshelman, and Das (1989)). The result of Schaffer et al. has also been formulated as the empirical expression

$$p_m \approx \frac{1.75}{\lambda\sqrt{l}} \tag{1}$$

where λ denotes the population size and l is the length of the individuals' genetic representation. Expression (1) is similar to that theoretically determined by Hesser and Männer (1991) for a special GA-variant

$$p_m(t) = \sqrt{\frac{\alpha}{\beta}} \cdot \frac{\exp\left(-\gamma \frac{t}{2}\right)}{\lambda\sqrt{l}} \tag{2}$$

which additionally introduces a time-dependency for p_m (α, β, γ are constants).

A time-dependency of mutation rates was first suggested by Holland (1975) himself, although he did not give a detailed choice of the parameter for the time-dependent reduction of p_m. Later on Fogarty (1989) used several time-dependent schedules for p_m, a measure which remarkably increased the GA-performance. Both approaches use a deterministic decrease of mutation rates over time, such that $\lim_{t\to\infty} p_m(t) = 0$. In addition, the mutation rate is handled as a global parameter, i.e. one parameter value of $p_m(t)$ is valid for all individuals of the population. A general mechanism for the adaptation of operator probabilities was presented by Davis (1989), who used the quality of the offspring generated by an operator as a measure to adapt its application probability. However, this technique runs into credit assignment problems similar to those identified by Wilson and Goldberg (1989) for Classifier Systems, when it is tried to reward operators that set the stage for a later production of high quality offspring. And even this technique uses a deterministic, global control rule for the alteration and adjustment of the operator probabilities[1].

The approach described here fundamentally differs from these mechanisms with respect to the following topics:

- Mutation rates are handled as temporal and individually differing parameters, which are incorporated into the genetic representation of the individuals (encoded as bitstrings).
- Mutation rates are also subject to mutation and selection, i.e. they undergo evolution as well as the object variables. No global, deterministic control for the alteration of mutation rates exists.
- Mutation rates are initialized at random.

As a result, p_m is no longer an external parameter of the GA, but it is subject to self-adaptation under certain circumstances (which will be explained later).

Technically, the implementation works as follows: For continuous parameter optimization problems of the form[2] $f : \prod_{i=1}^{n}[a_i, b_i] \to \Re$ either 1 or n encoded mutation rates are added to the genotype, which is now described as:

$$\begin{aligned} I &= \left(\{0,1\}^l \times \left(\{0,1\}^{\hat{l}}\right)^n\right) \\ &\cup \left(\{0,1\}^l \times \{0,1\}^{\hat{l}}\right) \end{aligned} \tag{3}$$

Here $\hat{l}$ denotes the length of the representation of a mutation rate. Furthermore, injective decoding functions $\Gamma_{a,b,l} : \{0,1\}^l \to [a, b]$ exist[3], which are mapping bitstrings of length l linearly to the real interval $[a, b]$. For $l = \sum_{i=1}^{n} l_i$ an individual $a \in I$ can be denoted as follows:

$$\begin{aligned} a &= (\alpha_{1,1} \ldots \alpha_{1,l_1} \ldots \alpha_{n,1} \ldots \alpha_{n,l_n}, \\ &\quad \hat{\alpha}_{1,1} \ldots \hat{\alpha}_{1,\hat{l}} \ldots \hat{\alpha}_{n,1} \ldots \hat{\alpha}_{n,\hat{l}}) \end{aligned} \tag{4}$$

Then, for the encoding $(\hat{\alpha}_{i,1} \ldots \hat{\alpha}_{i,\hat{l}})$ of the i-th mutation rate p_i, the mutation rate is obtained by $p_i = \Gamma_{0,1,\hat{l}}(\hat{\alpha}_{i,1} \ldots \hat{\alpha}_{i,\hat{l}})$. The new mutation mechanism $m : I \to I$ works as follows:

$$\begin{aligned} m((\alpha_{1,1} \ldots \alpha_{n,l_n}, \hat{\alpha}_{1,1} \ldots \hat{\alpha}_{n,\hat{l}})) = \\ (\beta_{1,1} \ldots \beta_{n,l_n}, \hat{\beta}_{1,1} \ldots \hat{\beta}_{n,\hat{l}}) \end{aligned} \tag{5}$$

where $\forall i \in \{1, \ldots, n\}\ \forall k \in \{1, \ldots, \hat{l}\}$:

$$\begin{aligned} \hat{\beta}_{i,k} &= \begin{cases} \hat{\alpha}_{i,k} & , \chi \le \Gamma_{0,1,\hat{l}}(\hat{\alpha}_{i,1} \ldots \hat{\alpha}_{i,\hat{l}}) \\ \kappa \in \{0,1\} & , \chi > \Gamma_{0,1,\hat{l}}(\hat{\alpha}_{i,1} \ldots \hat{\alpha}_{i,\hat{l}}) \end{cases} \\ \beta_{i,k} &= \begin{cases} \alpha_{i,k} & , \chi \le \Gamma_{0,1,\hat{l}}(\hat{\beta}_{i,1} \ldots \hat{\beta}_{i,\hat{l}}) \\ \kappa \in \{0,1\} & , \chi > \Gamma_{0,1,\hat{l}}(\hat{\beta}_{i,1} \ldots \hat{\beta}_{i,\hat{l}}) \end{cases} \end{aligned} \tag{6}$$

χ denotes a uniform random variable on the interval $[0, 1]$ which is sampled anew for each bit posi-

[1] An exception working with a really self-adapting crossover-operator is due to Schaffer and Morishima (1987), who encoded the crossover mechanism in the genotype of the individuals. This approach, although dealing with the operator itself instead of an application probability, is conceptually identical to the mechanism introduced here.

[2] Due to the binary encoding the object variables are in fact restricted to intervals $[a_i, b_i]$; $a_i, b_i \in \Re$.

[3] Let $\oplus$ denote summation modulo 2, and assume that a Gray code is used. Then, Γ often takes the form $\Gamma_{a,b,l}(\alpha_1 \ldots \alpha_l) = a + (b - a) \cdot \left(\sum_{i=0}^{l-1} \left(\bigoplus_{j=1}^{i+1} \alpha_j\right) 2^i\right) / \left(2^l - 1\right)$.

tion. Also $\kappa \in \{0,1\}$ is a uniform random variable, determining an allele value each time it is sampled[4]. The mutation mechanism first mutates the mutation rates p_i with mutation probabilities p_i and then uses the resulting mutation rates to mutate the corresponding object variable information. This is schematically shown in figure 1. The special case of only one mutation rate can also be used to apply the mechanism to discrete optimization problems. In this case, one mutation rate is valid for all bits of the genotype.

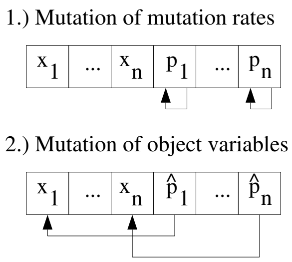

Figure 1: Schematic working mechanism of adaptive mutation

The asymptotic behaviour of adaptive mutation rates can be investigated when neither recombination nor selection are taken into account. To simplify notations, let $\hat{a} = (\hat{\alpha}_1 \ldots \hat{\alpha}_{\hat{l}})$ be the encoding of a mutation rate and $p_{\hat{a}} = \Gamma_{0,1,\hat{l}}(\hat{a})/2$ be the effective probability that a bit is changed. Then $p_{\hat{a}\to\hat{b}}$, the probability that mutation of $\hat{a}$ by using the bit-inversion probability $p_{\hat{a}}$ yields $\hat{b} = (\hat{\beta}_1 \ldots \hat{\beta}_{\hat{l}})$, is given by

$$p_{\hat{a}\to\hat{b}} = \prod_{i=1}^{\hat{l}} \Big(|\hat{\beta}_i - \hat{\alpha}_i| p_{\hat{a}} + (1 - |\hat{\beta}_i - \hat{\alpha}_i|)(1 - p_{\hat{a}}) \Big) \tag{7}$$

Some special transition probabilities can easily be obtained:

$$\begin{aligned} p_{\hat{a}\to\hat{a}} &= (1 - p_{\hat{a}})^{\hat{l}} \\ p_{(0\ldots0)\to\hat{b}} &= \prod_{i=1}^{\hat{l}} (1 - \hat{\beta}_i) = \begin{cases} 1 , & \hat{b} = (0\ldots0) \\ 0 , & \hat{b} \neq (0\ldots0) \end{cases} \end{aligned} \tag{8}$$

The set of $(2^{\hat{l}})^2$ possible transition probabilities forms a transition matrix of the corresponding markov chain. While $p_{\hat{a}\to\hat{b}}$ denotes the probability of a one–step transition, the probability $P_{\hat{a}\to\hat{b}}(n)$ to reach $\hat{b}$ from $\hat{a}$ after n transitions can be calculated by using a theorem from the theory of markov chains, e.g. to be found in the book by Gnedenko (1970):

$$P_{\hat{a}\to\hat{b}}(n) = \sum_{r=0}^{2^{\hat{l}}-1} P_{\hat{a}\to bin(r)}(m) \cdot P_{bin(r)\to\hat{b}}(n-m) \tag{9}$$

Here $bin : \{0,\ldots,2^{\hat{l}}-1\} \to \{0,1\}^{\hat{l}}$ denotes a mapping to binary representation. Then theorem 1 holds, which states that $(0\ldots0)$ is an absorbing state of the markov chain:

Theorem 1 (Asymptotic Behaviour)

$$\lim_{t\to\infty} P_{\hat{a}\to(0\ldots0)}(t) = 1 \quad \forall a \in \{0,1\}^{\hat{l}}$$

The proof is given in the appendix. Due to the absence of recombination on mutation rates and selection we can *not* conclude that convergence towards 0 is achieved by our algorithm (in fact, it should not be intended for mutation rates, because this would lead to the possibility of a reduction of the dimension of the actual search space).

While in ESs a meta-mutation rate is used to control the mutation of mutation rates (see Bäck, Hoffmeister, and Schwefel (1991) for an overview, Schwefel (1981) for an detailed description), experimental investigations using this mechanism for GAs have not been as successful as the mechanism described here.

An other precondition for self-adaptation to become effective concerns the selection operator. This will be discussed in the next section, before we present experimental results.

Extinctive Selection

A detailed classification of selection mechanisms in Evolutionary Algorithms has been given by Bäck and Hoffmeister (1991a). Here we will focus only on the topic of *extinctiveness*, which is important for self-adaptation of strategy parameters. Let a population at generation t be denoted as $P^t = (a_1^t, \ldots, a_\lambda^t)$ and let $p_s : I^\lambda \to [0,1]$ be the function determining the selection probabilities of the individuals in a population $(\forall P^t = (a_1^t, \ldots, a_\lambda^t) : \sum_{i=1}^{\lambda} p_s(a_i^t) = 1)$.

Definition 1 (Extinctive Selection)

A selection scheme is called *extinctive*:
$\Longleftrightarrow \ \forall t \geq 0 \ \ \forall P^t = (a_1^t, \ldots, a_\lambda^t) \ \ \exists i \in \{1,\ldots,\lambda\} :$ $p_s(a_i^t) = 0$

[4] Note, that following Holland (1975) a bit mutation event is realized here by chosing an allele value at random from the set $\{0,1\}$ of possible alleles, in contrast to the often used implementations which simply mutate a bit by inverting it.

DEFINITION 2 (Preservative Selection)

A selection scheme is called *preservative*:
$\iff \forall t \geq 0 \; \forall P^t = (a_1^t, \ldots, a_\lambda^t) \; \forall i \in \{1, \ldots, \lambda\} \; : \; p_s(a_i^t) > 0$

An extinctive selection mechanism definitely excludes some individuals from being selected, in contrast to *preservative* mechanisms which always assign selection probabilities greater than zero to all individuals. Representatives for preservative selection are *proportional selection* as introduced by Holland (1975) and Baker's (1985) *ranking*, while *(μ,λ)-selection* as used by Schwefel (1981) in ESs is extinctive (only the μ best individuals are allowed to be selected). The terms *(μ,λ)-proportional selection* and *(μ,λ)-linear ranking* are used here to describe the extinctive variants of proportional selection and ranking, while for (μ,λ)-selection the term *(μ,λ)-uniform ranking* is used. The latter term was introduced by Bäck and Hoffmeister (1991a) and emphasizes the fact, that (μ,λ)-selection is a special case of (μ,λ)-linear ranking selection.

Experimental Results

For testing the self-adaptive capabilities of the approach the objective functions f_1 (the sphere model, used by De Jong (1975) and Schwefel (1981)) and f_{15} (the weighted sphere model[5], used by Schwefel (1981)) as described in table 1 are used as representatives of relatively simple, unimodal problems. These functions make it possible to study whether the amount of mutation rate information that has to be learned should be different. Additionally, the multimodal function f_7 (generalized Rastrigin's function, based upon the two-dimensional variant mentioned by Törn and Žilinskas (1989), pp. 185) is used here in order to test adaptive mutation rates for a complicated problem.

A modified version of Grefenstette's (1987) GA-implementation GENESIS is used to obtain the experimental results, and general settings for each run of the GA are:

- Population size $\lambda = 50$.
- Length of the object variable encoding part of an individual $l = 32n$ (n being the dimension of the objective function[6]).
- Crossover rate $p_c = 0.6$.
- Two–point crossover, working only on the genotype encoding of the object variables.
- Gray code.

For adaptive mutation a length $\hat{l} = 20$ was chosen to allow sufficiently many different mutation rates to be encoded by the string. For each objective function a *reference* GA with (50,50)-proportional selection and a constant, external mutation rate $p_m = 0.001$ is compared to the following variants:

(1) (50,50)-proportional selection, 1 adaptive mutation rate.

(2) (50,50)-proportional selection, n adaptive mutation rates.

(3) (10,50)-proportional selection, no adaptive mutation ($p_m = 0.001$).

(4) (10,50)-proportional selection, 1 adaptive mutation rate.

(5) (10,50)-proportional selection, n adaptive mutation rates.

The setting of $\mu/\lambda = 1/5$ for the extinctive selection variants stems from theoretical results derived by Schwefel (1981) concerning the convergence velocity of ESs for the sphere and corridor model, and first experiments by Bäck and Hoffmeister (1991) indicated to use it even in the field of GAs. The experimental results are compared by looking at the best values per generation, which are averaged over 10 runs of the algorithm. In figures (2)–(4) the resulting graphs are shown for f_1, f_{15}, and f_7, respectively. The plots are labeled by their numbers given in the enumeration above, the reference GA being labeled with (0).

First we will look at the combination of preservative selection and adaptive mutation, i.e. cases (1) and (2). For the unimodal functions (figures 2 and 3) performance decreases as the amount of additional information increases, while for the multimodal function (figure 4) one mutation rate (1) converges to a better local optimum than the reference GA, and n mutation rates (2) improve slowly but steadily, possibly being better than (0) and (1) on the long run. Thus, without changing the selection mechanism the large amount of additional information is disadvantageous

[5] The index choice for objective functions is founded on the historical development of our implementation as well as the attempt to keep in correspondence with De Jongs (1975) nomenclature f_1–f_5.

[6] A length of 32 bits per object variable is used for the representation of the real interval $[a, b]$ to which the bitstrings are mapped, in order to achieve a maximum resolution $\Delta x = (b - a)/(2^{32} - 1)$ of the search grid.

Name	Description	Dim.	Characteristics	Ref.
f_1	sphere model $f_1(\vec{x}) = \sum_{i=1}^{n} x_i^2$ $-5.12 \leq x_i \leq 5.12$	$n = 30$	unimodal, high–dimensional	De Jong (1975) Schwefel (1981)
f_{15}	weighted sphere model $f_{15}(\vec{x}) = \sum_{i=1}^{n} i \cdot x_i^2$ $-5.12 \leq x_i \leq 5.12$	$n = 30$	unimodal, high–dimensional	Schwefel (1988)
f_7	generalized Rastrigin's function $f_7(\vec{x}) = nA + \sum_{i=1}^{n} x_i^2 - A\cos(\omega x_i)$ $A = 10\ ; \omega = 2\pi\ ; -5.12 \leq x_i \leq 5.12$	$n = 20$	multimodal, high–dimensional, f_1 with sine wave superposition	Törn and Žilinskas (1989)

Table 1: The set of test functions

for unimodal and of slight advantage for multimodal topologies.

When introducing extinctive selection (3) alone, the behaviour is contrary. On unimodal functions performance increases remarkably, while on the multimodal function extinctive selection does not change performance when compared to the reference GA.

In any case the GA remarkably benefits from the combination of adaptive mutation and extinctive selection, although it accomplishes this behaviour in different combinations with respect to the number of mutation rates. The unimodal functions are better suited for search with extinctive selection and only one mutation rate (4) than n rates (5), while for the multimodal function one mutation rate quickly converges to a local optimum and n mutation rates lead to the best solution with a population not yet converged after 500 generations.

However, these results for unimodal functions are contradicting to those obtained by Schwefel (1988) for self-adaptation of mutation rates in an ES. For f_1 and f_{15} he found remarkably better behaviour for variants using a combination of individual step sizes and recombination than for variants using only one step size. This synergetic effect is mainly caused by the recombination of mutation rates, which supports the emergence of actually better suited combinations of mutation rates. To prevent the mechanism from reducing the search space by a mutation rate which becomes zero, in ESs Schwefel (1981) uses an additional exogeneous multiplication factor for all standard deviations of an individual. By using a log-normal distribution for this factor its expectation is 1 and the occurrence probability of a multiplication factor r is equal to that of a factor $1/r$, thus a subspace search is prevented by such a mechanism.

The GA-behaviour found in these experiments is a clear confirmation of the well-known contradiction between exploration and exploitation in global optimization. For unimodal surfaces a path oriented, exploitative search with a high convergence velocity is desired, while for multimodal surfaces the search should be volume oriented, explorative with a high convergence confidence towards a global optimum point. Both property classes are contradicting, one can not have them at the same time.

Applied to the results given here, only one adaptive mutation rate seems to serve as a local hillclimbing mechanism when combined with extinctive selection. On the other hand, a high degree of exploration is achieved when all object variables are mutated independently, thus serving as a powerful global exploration mechanism when combined with a strong emphasis on survival of the fitter individuals to prevent the algorithm from converging too slow.

In case (4) of one adaptive mutation rate figure 5 gives an impression of the course of the average mutation rate over time. Two facts are interesting to note:

- For extinctive selection average mutation rates are almost a factor of 10 larger than for preservative selection.
- There is no difference in the average mutation rates between the multimodal and the unimodal functions.

From these observations we can conclude that the algorithm balances well between a mutation rate as high as needed for efficient search and as high as possible without destroying useful information, thus working on the borderline of efficient optimization and almost random walk.

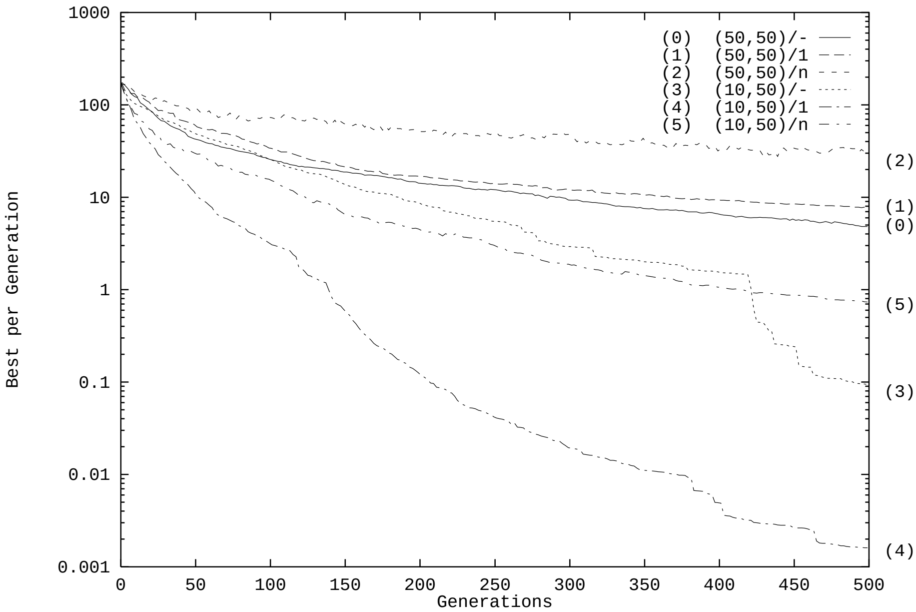

Figure 2: Performance on f_1

Understanding the mutation rates of figure 5 as optimal ones, we can compare them to the results obtained by Schaffer's empirical expression (1), which are

- $p_m = 0.0011$ for f_1 and f_{15} ($\lambda = 50$, $l = 960$),
- $p_m = 0.0014$ for f_7 ($\lambda = 50$, $l = 640$).

Unfortunately the results can not be compared directly due to the fact that the mutation rates used here are probabilities for sampling a bit position instead of inverting it, and Schaffer et al. do not explain which kind of mutation they use. However, at the most our mutation rates would double, increasing the gap between the results. We can conclude by formulating hypothesis 1:

Hypothesis 1

An optimal mutation rate for a GA is relatively large and turns mutation into an additional search operator. When used in combination with extinctive selection, it considerably improves the performance of a GA.

Concerning the GA this hypothesis is a daring one. Holland (1975, pp. 110–111) has explicitly stressed the role of mutation as a "*background operator*", operating as a mechanism to reintroduce lost alleles, but *not* as a search operator. On the other hand, Eigen (1976) has demonstrated that for more general models of evolving systems on the level of organic macromolecules evolution works most efficient when mutation rates are directly below the threshold value above which information is destroyed. Although the investigations of Eigen do not take recombination into account, they are theoretically well-founded and provide a strong argument for enlarging the role of mutation even in the artificial evolving systems we are dealing with.

Summary

Within this work it is confirmed that under the condition of an extinctive selection mechanism second-level learning of mutation rates is possible and advantageous even in GAs. For a multimodal function n mutation rates per individual perform well, while for unimodal functions one mutation rate per individual performs well. In any case at least a combination of one adaptive mutation rate per individual and extinctive selection is better than the reference GA. This is a strong argument for the general introduction of adaptive mutation rates to GAs, which would also eliminate a part of the parameterisation problem of GAs. However some questions haven't been answered yet, especially concerning recombination of mutation rates, which has shown by Schwefel (1988) to be an

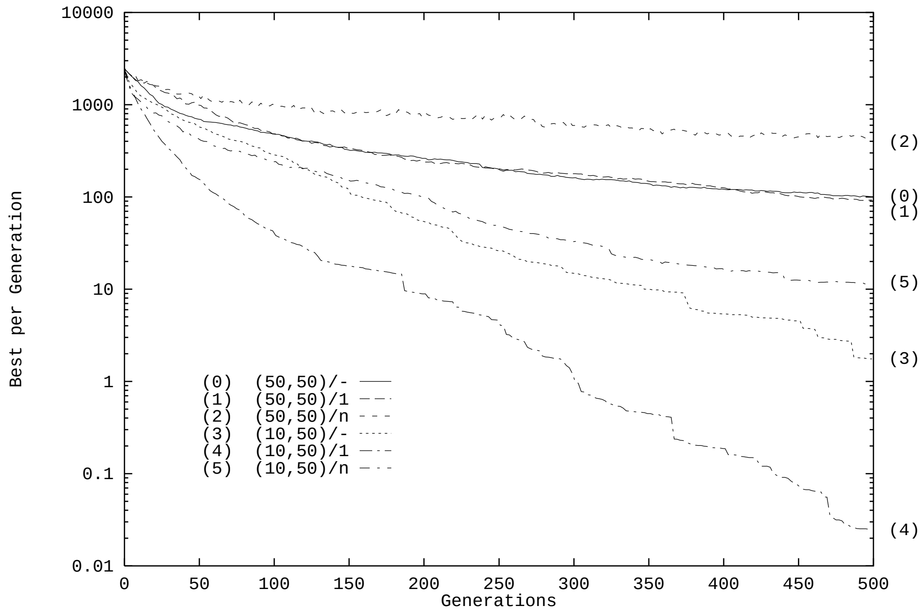

Figure 3: Performance on f_{15}

essential condition[7] for self-adaptation in ESs. Surely, new recombination techniques have to be used for n mutation rates, while for one mutation rate traditional crossover could be used.

In addition the influence of different extinctive selection mechanisms, especially those which are rank-based, as well as the influence of the ratio μ/λ, i.e. the degree of extinctiveness, should be investigated further.

Appendix: Proof of theorem 1

The proof of theorem 1 is based upon equation (9) for calculating the n-step transition probabilities of the markov chain:

$$
\begin{aligned}
& P_{a\rightarrow(0\ldots0)}(n) \\
&= \sum_{r=0}^{2^{lm}-1} P_{a\rightarrow bin(r)}(m)\, P_{bin(r)\rightarrow(0\ldots0)}(n-m) \\
&= \sum_{r=0}^{2^{lm}-1} P_{a\rightarrow bin(r)}(n-1)\, P_{bin(r)\rightarrow(0\ldots0)}(1) \\
&\quad \text{for } m = n-1 \\
&= P_{a\rightarrow(0\ldots0)}(n-1)\, P_{(0\ldots0)\rightarrow(0\ldots0)}(1) \\
&\quad + \sum_{r=1}^{2^{lm}-1} P_{a\rightarrow bin(r)}(n-1)\, P_{bin(r)\rightarrow(0\ldots0)}(1) \\
&= P_{a\rightarrow(0\ldots0)}(n-1) \\
&\quad + \sum_{r=1}^{2^{lm}-1} P_{a\rightarrow bin(r)}(n-1)\, P_{bin(r)\rightarrow(0\ldots0)}(1) \\
&\geq P_{a\rightarrow(0\ldots0)}(n-1)
\end{aligned}
$$

[7] Another essential condition for self-adaptation in ESs is the possibility to forget good information. Schwefel (1988) has demonstrated that for this reason self-adaptation does not work in a $(\mu+\lambda)$-ES, which allows arbitrarily long survival of individuals. A standard GA without the *generation gap* by Grefenstette (1986) always allows to die even for the best individual, hence we do not discuss this topic here in detail for GAs.

Hence, $P_{a\rightarrow(0\ldots0)}$ increases monotonously over time. Since $\sum_{r=0}^{2^{lm}-1} P_{a\rightarrow bin(r)}(n) = 1$ and $P_{a\rightarrow(0\ldots0)}(n) \leq 1$, either

$$\sum_{r=1}^{2^{lm}-1} P_{a\rightarrow bin(r)}(n-1)\, P_{bin(r)\rightarrow(0\ldots0)}(1) > 0$$

or

$$P_{a\rightarrow(0\ldots0)}(n-1) = 1$$

must be valid ($P_{bin(r)\rightarrow(0\ldots0)}(1) = p_{bin(r)\rightarrow(0\ldots0)} \neq 0$ iff $r \neq 0$), hence either the probability of a transition to zero increases strictly, i.e. $P_{a\rightarrow(0\ldots0)}(n) > P_{a\rightarrow(0\ldots0)}(n-1)$, or $P_{a\rightarrow(0\ldots0)}(n) = 1$. q.e.d.

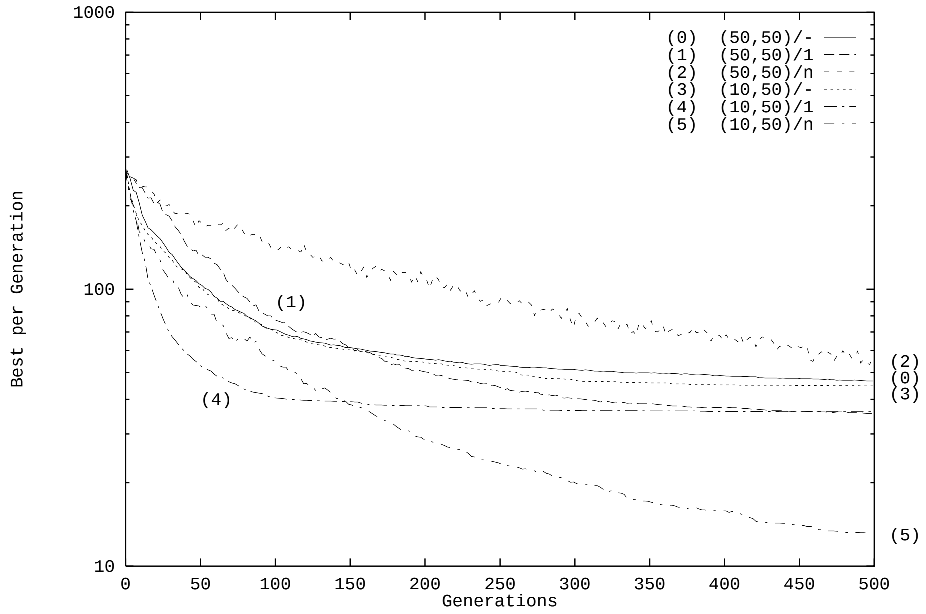

Figure 4: Performance on f_7

References

Bäck, T. and F. Hoffmeister. 1991a. Extended selection mechanisms in genetic algorithms. In *Proceedings of the Fourth International Conference on Genetic Algorithms and their Applications*, edited by Richard K. Belew and Lashon B. Booker, pages 92–99. University of California, San Diego, USA. Morgan Kaufmann Publishers.

Bäck, T. and F. Hoffmeister. 1991b. Global optimization by means of evolutionary algorithms. In *Random Search as a Method for Adaptation and Optimization of Complex Systems*, edited by A. A. Antamoshkin, pages 17–21. Krasnojarsk Space Technology University, Divnogorsk, USSR.

Bäck, T., F. Hoffmeister, and H.-P. Schwefel. 1991. A survey of evolution strategies. In *Proceedings of the Fourth International Conference on Genetic Algorithms and their Applications*, edited by Richard K. Belew and Lashon B. Booker, pages 2–9. University of California, San Diego, USA. Morgan Kaufmann Publishers.

Baker, J. E. 1985. Adaptive selection methods for genetic algorithms. In *Proceedings of the First International Conference on Genetic Algorithms and Their Applications*, edited by J. J. Grefenstette, pages 101–111. Hillsdale, New Jersey. Lawrence Erlbaum Associates.

Davis, L. 1989. Adapting operator probabilities in genetic algorithms. In *Proceedings of the Third International Conference on Genetic Algorithms and Their Applications*, edited by J. D. Schaffer, pages 61–69. San Mateo, California. Morgan Kaufmann Publishers.

Eigen, M. 1976. Wie entsteht Information? Prinzipien der Selbstorganisation in der Biologie. *Berichte der Bunsen-Gesellschaft*, 11:1059–1081.

Fogarty, T. C. 1989. Varying the probability of mutation in the genetic algorithm. In *Proceedings of the Third International Conference on Genetic Algorithms and Their Applications*, edited by J. D. Schaffer, pages 104–109. San Mateo, California. Morgan Kaufmann Publishers.

Gnedenko, B. W. 1970. *Lehrbuch der Wahrscheinlichkeitsrechnung.* Akademie–Verlag, Berlin.

Grefenstette, J. J. 1986. Optimization of control parameters for genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics*, SMC–16(1):122–128.

Grefenstette, J. J. 1987. *A User's Guide to GENESIS.* Navy Center for Applied Research in Artificial Intelligence, Washington, D. C.

Hesser, J. and R. Männer. 1991. Towards an optimal mutation probability in genetic algorithms. In *Parallel Problem Solving from Nature*, edited by Hans-Paul Schwefel and Reinhard Männer, pages 23–32. Volume

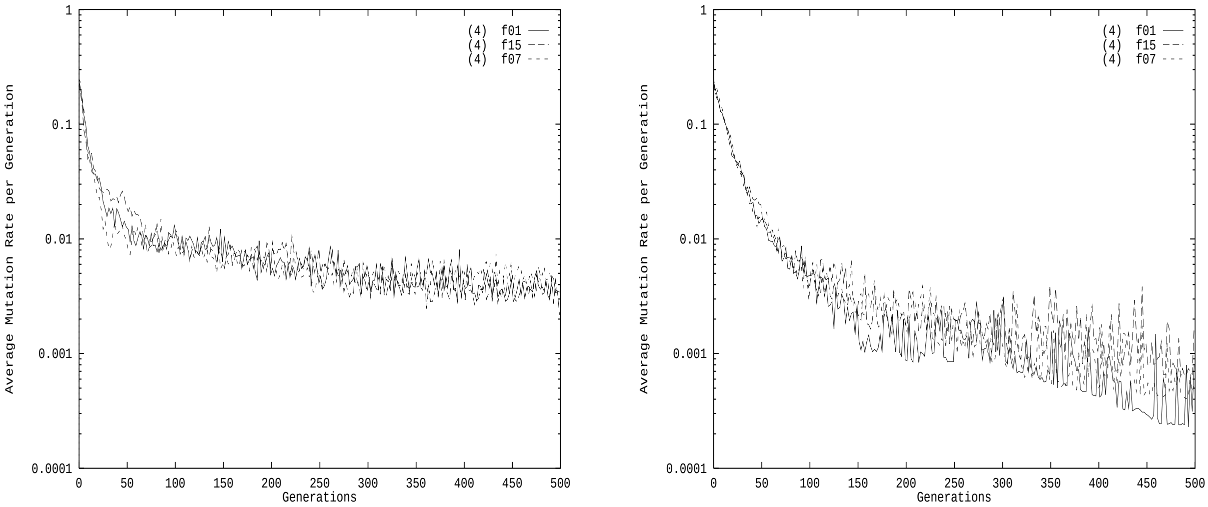

Figure 5: Average mutation rates for (10,50)- (left) and (50,50)-proportional selection (right)

496 of *Lecture Notes in Computer Science*, Springer, Berlin.

Hoffmeister, F. and T. Bäck. 1991. Genetic algorithms and evolution strategies: Similarities and differences. In *Parallel Problem Solving from Nature*, edited by Hans-Paul Schwefel and Reinhard Männer, pages 455–470. Volume 496 of *Lecture Notes in Computer Science*, Springer, Berlin.

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

De Jong, K. 1975. *An analysis of the behaviour of a class of genetic adaptive systems.* PhD thesis, University of Michigan. Diss. Abstr. Int. 36(10), 5140B, University Microfilms No. 76–9381.

Langton, C. G. 1989. Artificial life. In *Artificial Life*, edited by Christopher G. Langton, pages 1–48. Volume VI of *Studies in the Science of Complexity*, Los Alamos, Santa Fe Institute. Addison-Wesley.

Mühlenbein, H. 1991. Evolution in time and space — the parallel genetic algorithm. In *Foundations of Genetic Algorithms & Classifier Systems*, edited by G. Rawlins. Morgan Kaufmann.

Rechenberg, I. 1973. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.* Frommann–Holzboog Verlag, Stuttgart.

Schaffer, J. D., R. A. Caruna, L. J. Eshelman, and R. Das. 1989. A study of control parameters affecting online performance of genetic algorithms for function optimization. In *Proceedings of the Third International Conference on Genetic Algorithms and Their Applications*, edited by J. D. Schaffer, pages 51–60. San Mateo, California. Morgan Kaufmann Publishers.

Schaffer, J. D., and A. Morishima. 1987. An adaptive crossover distribution mechanism for genetic algorithms. In *Proceedings of the Second International Conference on Genetic Algorithms and Their Applications*, edited by J. J. Grefenstette, pages 36–40. Hillsdale, New Jersey. Lawrence Erlbaum Associates.

Schwefel, H.-P. 1981. *Numerical Optimization of Computer Models.* Wiley, Chichester.

Schwefel, H.-P. 1988. Evolutionary learning optimum–seeking on parallel computer architectures. In *Proceedings of the International Symposium on Systems Analysis and Simulation 1988, I: Theory and Foundations*, edited by A. Sydow, S. G. Tzafestas, and R. Vichnevetsky, pages 217–225. Akademie der Wissenschaften der DDR, Akademie-Verlag, Berlin.

Törn, A. and A. Žilinskas. 1989. *Global Optimization*, volume 350 of *Lecture Notes in Computer Science.* Springer, Berlin.

Wilson, S. W., and D. E. Goldberg. 1989. A critical review of classifier systems. In *Proceedings of the Third International Conference on Genetic Algorithms and Their Applications*, edited by J. D. Schaffer, pages 244–255. San Mateo, California. Morgan Kaufmann Publishers.