An Alternative Explanation for the Manner in which Genetic Algorithms Operate

Hans-Georg Beyer
University of Dortmund
Department of Computer Science
Systems Analysis Research Group
D-44221 Dortmund
Germany

beyer@LS11.informatik.uni-dortmund.de

July 3, 1996

Abstract

The common explanation of the manner in which genetic algorithms (GAs) process individuals in a population of contending solutions relies on the "building block hypothesis." This suggests that successively better solutions are generated by combining useful parts of extant solutions. An alternative explanation is presented which focuses on the collective phenomena taking place in populations that undergo recombination. The new explanation is derived from investigations in evolution strategies (ESs). The principles studied are general, and hold for all evolutionary algorithms (EAs), including genetic algorithms (GAs). Further, they appear to be somewhat analogous to some theories and observations on the benefits of sex in biota.

Keywords

building block hypothesis, evolutionary algorithms, multi-recombination

1 Introduction

Although specific theoretical investigations into the properties of genetic algorithms have gained considerable recent attention, there still is no satisfactory theory that describes the basic working principles of these methods. Moreover, there is no generally accepted theory that would seek to explain the precise working of all variations of the evolutionary computation, including genetic algorithms, evolution strategies, and evolutionary programming. There is debate on the importance of mutation and recombination, and in particular crossover (see Fogel and Atmar, 1990; Spears, 1992; Fogel and Stayton, 1994). And the importance of many other facets of evolutionary computation, including *deception*, the *schema theorem*, and *implicit parallelism*, have been argued (e.g., Grefenstette, 1991; Whitley, 1991; De Jong, 1993; Grefenstette, 1993; Altenberg, 1995). Current analysis within genetic algorithms does not provide a framework for predicting the influence of various procedures and operators. Such analysis has been conducted within evolution strategies, however (Beyer, 1995ab). The results of this analysis should be useful for interpreting the interaction between mutation and recombination, and at first glance, the results are unexpected and are diametrically opposed to the *building block hypothesis* (Goldberg, 1989).

The building block hypothesis states that genetic algorithms generate successful solutions to optimization problems by combining useful parts of those solutions. This general procedure has been described as "adaptation by credit assignment" in Holland (1995, p. 53). Yet the results of Beyer (1995ab) (described below) do not confirm this notion of the utility of recombination, and this raises a question as to whether or not the schema theorem and implicit parallelism (Holland, 1975) are really the essential essences of a genetic algorithm (GA).

This article is organized as follows. First, a short discussion about the questions just raised and the status of current genetic algorithm theories is given in section 2. Section 3 is devoted to the $(\mu/\mu, \lambda)$ evolution strategies (ESs). This class of strategies performing multirecombination and (μ, λ)-truncation selection exhibits larger progress rates than its non-recombinative counterpart, the (μ, λ)-ES (theoretically provable). The basic working principles will be extracted in order to explain the benefit of recombination. In section 4 we will discuss the findings made for the $(\mu/\mu, \lambda)$-ES in the light of GAs.

2 Current Theories of Genetic Algorithms

2.1 The Schema Theorem

Holland (1975) proposed examining collections of templates can be used to describe potential solutions. For example, over the alphabet of symbols $\{\mathtt{0}, \mathtt{1}\}$ with the additional wild-card symbol "?", the four-bit template [10??] can be used to refer to the four strings $\mathbf{b}_i$: [1000], [1001], [1010], and [1011]. Each such template is described as a schema.[1] The schema theorem generally states that under reproduction in proportion to relative fitness,

[1]From mathematical point of view, "schemata" are nothing else but *subsets* of the binary space $\{\mathtt{0}, \mathtt{1}\}^\ell$ (ℓ - length of the binary string). They can be illustrated by hyperplane partitions of the ℓ-dimensional

the expected number of instances of a particular schema $\mathcal{S}$ in a population at generation $g+1$

$$M(\mathcal{S}, g+1) = \frac{\sum_{\mathbf{b}\in\mathcal{S}} F(\mathbf{b})}{n(\mathcal{S})\,\overline{F}}\, M(\mathcal{S}, g) \tag{1}$$

where $M(\mathcal{S}, g)$ is the number of instances of $\mathcal{S}$ in the population and generation g, $F(\mathbf{b})$ is the fitness function, $n(\mathcal{S}) = \sum_{\mathbf{b}\in\mathcal{S}} 1$ is the number of strings that contain the schema $\mathcal{S}$, and $\overline{F}$ is the mean fitness of all strings in the population. Goldberg (1989, p. 33) offered "Short, low-order, above-average schemata receive exponentially increasing trials in subsequent generations." Although this statement is not wrong when applied to each successive generation, it is only a weak predictor of the behavior of a GA. Crossover and mutation do not appear in Eq. 1, and thus the theorem only describes the propagation of schemata that already exist in the population. When the effects of crossover and mutation are included in Eq. 1, they are viewed as disturbances and the equation must be rewritten as an inequality (Holland, 1992, p. 111) expressing a lower bound on the expected number of schema $\mathcal{S}$ in generation $g+1$. But even when these effects are not considered, the schema theorem cannot well predict the behavior of a GA over successive generations because (1) the equation holds for infinite populations only, and (2) sampling errors may result in schemata with observed average fitness that does not correspond to their actual average fitness. Moreover, from the perspective of function optimization, the schema theorem does not provide any guarantee of convergence to, or divergence from, any optimum solution. This is a significant weakness of the theorem.

The *building block hypothesis* comes in close connection with the schema theorem. Schemata are viewed as building blocks that may be useful in constructing complete solutions. Goldberg (1989, p. 41) asserts: "Just as a child creates magnificent fortresses through the arrangement of simple blocks, so does a genetic algorithm seek near optimal performance through the juxtaposition of short, low-order, high-performance schemata, or building blocks." This assertion is not incorrect, if taken only as an empirical observation from the optimization process. But if the hypothesis is regarded as a rationale for how the GA processes solutions then cause and effect are *reversed*. Furthermore, the observations that lead to the hypothesis are not features that are exclusive to GAs. They hold for any iterative procedure that successively approaches an optimum, if building blocks are interpreted as certain decompositions of the optimum solution. The exponential growth of schemata can even be observed in gradient strategies under such conditions.

Also in close connection with the schema theorem is the notion of implicit parallelism, which highlights that a sample from a schema such as [`10??`] above is implicitly sampling from many schemata, including that listed above and also [`10?0`], [`10?1`], [`100?`], and [`101?`]. The traditional efforts in GAs utilized binary coding schemes so as to maximize the number of schemata being implicitly testing in parallel in this manner (Holland, 1975). Goldberg (1989, p. 40) offered: "Even though each generation we perform computation proportional

hypercube where each vertex represents one of the 2^ℓ possible states of the individuals.

to the size of the population, we get useful processing of something like n^3 schemata in parallel ...", and also (p. 20): "Because this processing leverage is so important (and apparently unique to genetic algorithms), we give it a special name, *implicit parallelism*."

Leaving aside that the n^3-counting argument is questionable (cf. Grefenstette, 1991), implicit parallelism is certainly not unique to GAs. The counting argument is mainly based upon the number of schemata owned by an individual string of length ℓ. Each individual is a member of exactly 2^ℓ schemata and a population of μ individuals contains up to $\mu 2^\ell$ schemata (theoretical upper bound). However, consider a gradient-like strategy (often referred to as hillclimbing). Each data point needed to compute the gradient direction is a member of 2^ℓ schemata too! Even a gradient strategy would exhibit some kind of implicit parallelism.

Neither the building block hypothesis nor the notion of implicit parallelism suffices to explain the manner in which GAs process trials, nor do they explain features that are unique to GAs.

2.2 Markov Chain Analysis

Markov chains provide an exact microscopic model of GAs. Analysis to date has concentrated mainly on convergence properties (i.e., for $g \to \infty$, g – generation counter). Approaches to investigate transient behavior exist (e.g. De Jong et al., 1995), but they have not been broadly insightful so far. In terms of optimization, the crucial interest is the convergence rate (and order) to an optimum point, not to any steady state distribution in probability. One exact result from the Markov chain analysis is, that the GA offered by Holland (1975) does not provide convergence to local or global optima (Davis and Principle, 1993; Rudolph, 1994). This can be easily explained if one takes into account that this GA uses fitness-proportional selection and therefore always has a nonzero probability for losing any current solution. Convergence, however, can be obtained and proved if some form of elitism (maintaining the best solution found) is incorporated (De Jong, 1992; Rudolph, 1994). But this still leaves the question of how fast does such an algorithm approach an optimum. Although Markov chain analysis has showed that, in terms of asymptotic behavior, the inclusion of crossover does not bring any advantage to the GA (Davis and Principle, 1993), there has been no success in using it to determine rates of convergence in problem classes. However, this is the prerequisite if one wants to compare the influence of the operators on the performance of the algorithm.

The full statistical dynamics (i.e. the time evolution) of the microscopic Markov chain contains all information of the EA-system (EA – evolutionary algorithm)

$$\text{EA-system} := \text{EA} \;\;\&\;\; \text{fitness function}$$

This information is difficult to be interpreted and of minor interest. What is really of interest – as far as optimization is considered – is the dynamics of certain average values (mean value dynamics, see below) of the population toward the optimum. Such an information can be used to estimate the time complexity of the algorithm, i.e., the number of generations or fitness evaluations needed to reach the vicinity of the optimum.

2.3 The Mesoscopic Approach

Deriving analytical formulae for the mean value dynamics and the time complexity from the microscopic Markov chain are up to now an open problem, even for the simplest EA-systems. The most promising approach in the GA-field may be by an analysis which might be called "mesoscopic." It relies on features from the exact microscopic theory, but uses sometimes phenomenological "ingredients" obtained by empirical methods.

One way of bypassing the complicated Markov chain model is to introduce an infinite population model. Vose and Liepins (1991) introduced a mathematical formulation of the GA that approximates the population by a probability distribution of finding a single individual in a certain state (state occupancy probability). This model has been used to investigate certain dynamic properties of the standard GA (e.g., the observed rapid fluctuations of population states that may be akin to the idea of punctuate equilibrium). Recently, Vose and Wright (1995) provided the first results concerning the dynamics of the expected average population fitness on a class of fitness functions. Their results can be used as approximations for finite populations.

Another approach relying on methods from quantitative genetics has been taken by Mühlenbein and Schlierkamp-Voosen (1994) to investigate the time complexity of (μ, λ)-ES-like GAs (so-called breeder GAs, for the definition of (μ, λ)-ESs, see below) on simple test functions. Especially the bit-counting function (`OneMax`) has been investigated

$$\texttt{OneMax:} \quad F(\mathbf{b}) = \sum_{i=1}^{\ell} b_i. \tag{2}$$

The ℓ bit positions b_i can be treated independently from each other. This should simplify the analysis considerably. In spite of this, the *analytical* calculation of the average fitness change over the generations g for a GA, including the three operators selection, mutation, and crossover (not as single operators but their coupling), is still a pending problem.

3 Basic Principles from the $(\mu/\mu, \lambda)$-Evolution Strategy Theory

3.1 The Evolution Strategy Algorithms

Current ES versions are multimembered strategies that operate with a population (or populations). They are characterized by the (μ, λ)-nomenclature (see e.g. Schwefel, 1995) indicating that a population of μ parents at time g (g is the generation counter) generate λ offspring for the time step $g+1$. The μ parents are obtained from the offspring at generation g by truncation selection. That is, given the λ offspring $\mathbf{y}_l^{(g)}$ ($\mathbf{y}$ is usually a vector with real-valued, integer, or binary components) the μ best individuals with respect to their fitness $F(\mathbf{y})$ (the function to be optimized) are chosen to serve as the parents of the new generation. This selection process can be expressed by the "$m;\lambda$"-nomenclature, which simply means 'take the mth best out of the λ'. Thus, the parents of the (μ, λ)-selection are given by $\mathbf{y}_{m;\lambda}^{(g)}$, where $m = 1 \dots \mu$. After the production of the parental pool $\{\mathbf{y}_{1;\lambda}^{(g)}, \mathbf{y}_{2;\lambda}^{(g)}, \dots \mathbf{y}_{\mu;\lambda}^{(g)}\}$ the "genetic" operator(s) mutation (and recombination) is (are) applied. The concrete realization of these operators depends on the problem domain considered (continuous parameter optimization, integer programming, pseudo-boolean optimization, combinatorial optimization).

The relevant theory is associated with – but not restricted to – the area of real-valued N-dimensional parameter optimization. Mutations in this domain are generated by a vector variation operator $\mathbf{z}$ which is usually Gaussian distributed

$$\mathbf{z} = (z_1, z_2, \dots z_N)^{\mathrm{T}} := \mathcal{N}(\mathbf{0}, \mathbf{C}), \tag{3}$$

with zero mean and covariance matrix $\mathbf{C}$. For now, let $\mathbf{C}$ be $\sigma^2\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Such mutations are referred to as isotropic Gaussian mutations with mutation strength σ. There is no preference for a certain direction in the state space of the $\mathbf{y}$ vectors (unbiased mutations). Each component z_i of $\mathbf{z}$ obeys the normal density $p(z_i)$

$$p(z_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{z_i^2}{\sigma^2}\right). \tag{4}$$

Using the mutations introduced above, we are able now to express the ES algorithms in a compact fashion. The simplest ES variant, denoted as (μ, λ)-ES is without recombination

$$\forall\, l = 1 \dots \lambda: \quad \underline{(\mu, \lambda)\text{-ES:}} \quad \begin{cases} \mathbf{y}_l^{(g+1)} := \mathbf{y}_{m;\lambda}^{(g)} + \mathbf{z}, \\ m := \mathtt{Random}\{1 \dots \mu\} \end{cases} \tag{5}$$

That is, each offspring is generated from the parental pool by picking up uniformly at random one of the parents and subsequent addition of an isotropic random vector $\mathbf{z}$. All components of $\mathbf{z}$ are generated with equal standard deviation σ. The mutation strength σ

plays a similar role as the bit-flipping probability in GAs. The performance of the ES can be varied by adjusting σ.

A *multi-mixing strategy* for variation can be imposed during the creation of offspring in addition to other forms of recombination and mutation. Such an operation is denoted by $(\mu/\rho, \lambda)$, where ρ indicates the number of parents involved in the act of procreation. Here, attention is restricted to the case where $\rho = \mu$. There are two variants of this multirecombination.

Intermediate recombination, denoted by $(\mu/\mu_I, \lambda)$, is equivalent to a center of mass of the collection of μ parents $\{\mathbf{y}_{m;\lambda}\}$

$$\langle \mathbf{y} \rangle_\mu := \frac{1}{\mu} \sum_{m=1}^{\mu} \mathbf{y}_{m;\lambda}, \tag{6}$$

and random mutation is imposed on top of this center of mass. Thus, the offspring $\mathbf{y}_l$ are created by

$$\forall\, l = 1 \ldots \lambda : \quad \underline{(\mu/\mu_I, \lambda)\text{-ES:}} \quad \mathbf{y}_l^{(g+1)} := \frac{1}{\mu} \sum_{m=1}^{\mu} \mathbf{y}_{m;\lambda}^{(g)} + \mathbf{z}. \tag{7}$$

The *dominant recombinant* $(\mu/\mu_D, \lambda)$-ES generates the ith component of an offspring $\mathbf{y}_l^{(g+1)}$, denoted by $(\mathbf{y}_l^{(g+1)})_i$, by randomly choosing one of the ith components from the parental pool and additive application of the $z_i = \mathcal{N}(0, \sigma^2)$ mutation

$$\forall\, l = 1 \ldots \lambda,\ i = 1 \ldots N : \quad \underline{(\mu/\mu_D, \lambda)\text{-ES:}} \quad \begin{cases} \left(\mathbf{y}_l^{(g+1)}\right)_i := \left(\mathbf{y}_{m_i;\lambda}^{(g)}\right)_i + \mathbf{z}_i, \\ m_i := \mathtt{Random}\{1 \ldots \mu\}. \end{cases} \tag{8}$$

This is equivalent to Schwefel's global, discrete recombination (see Schwefel and Rudolph, 1995) and reminiscent of the uniform crossover in GAs (Syswerda, 1989) which applies to only two parents.

3.2 Measuring Performance

A natural performance measure for function optimizers is the distance-to-optimum change ΔR observed from generation g to $g + 1$. The expectation taken over ΔR is called the progress rate $\varphi := E[\Delta R]$. For a population, there are different possible definitions of the distance-to-optimum. Usually, the average distance from the μ parents to the optimum point $\hat{\mathbf{y}}$ is taken. If $\hat{\mathbf{y}}$ is positioned at the origin then

$$\varphi \ := \ E\left[R \ - \frac{1}{\mu} \sum_{m=1}^{\mu} \left\| \mathbf{y}_{m;\lambda}^{(g+1)} \right\| \right], \tag{9}$$

with

$$R = \frac{1}{\mu} \sum_{m=1}^{\mu} \left\| \mathbf{y}_{m;\lambda}^{(g)} \right\|. \qquad (10)$$

The progress rate φ depends on the landscape defined by fitness function $F(\mathbf{y})$. In order to obtain analytical φ-formulae, $F(\mathbf{y})$ must appropriately be chosen. Furthermore one expects that this model function should be capable of approximating locally more complex fitness landscapes. This holds for the spherical model, i.e., $F(\mathbf{y})$ depends on the radius function $R = \|\mathbf{y}\|$ only.

The analysis on the sphere model is scalable, i.e., the progress rate φ can be expressed without regard to the actual $R^{(g)}$ and N (the parameter space dimension), if the normalized progress rate φ^* and the normalized mutation strength σ^* are introduced

$$\varphi^* := \varphi \frac{N}{R}, \qquad \sigma^* := \sigma \frac{N}{R}. \qquad (11)$$

Below, the main results of the progress rate theory on the sphere model will be presented. The main emphasis is on the extraction and the explanation of the basic principles. For this reason the asymptotic progress rate formulae for $N \to \infty$ suffice. The treatment of the $N < \infty$ case can be found in Beyer (1995ab).

3.3 Basic Principles

3.3.1 The (μ, λ)-Theory

For the case $N \to \infty$,

$$\varphi^* = c_{\mu,\lambda}\sigma^* - \frac{(\sigma^*)^2}{2}. \qquad (12)$$

The influence of the selection is described by the so-called progress coefficient $c_{\mu,\lambda}$ (Beyer, 1995a) bounded by

$$0 \le c_{\mu,\lambda} \le c_{1,\lambda} \le \sqrt{2 \ln \lambda}. \qquad (13)$$

Positive progress, i.e., convergence to the optimum is achieved for σ^*-values smaller than $2c_{\mu,\lambda}$. The maximal (normalized) progress is $(c_{\mu,\lambda})^2/2$, reachable if σ^* is tuned such that $\sigma^* = c_{\mu,\lambda}$. If renormalized by Eq. 11, $\sigma = c_{\mu,\lambda}R/N$ indicates that approaching the optimum (R decreases) the mutation strength has to be decreased

$$g \to \infty \;\Rightarrow\; \sigma \to 0. \qquad (14)$$

Usually, this is attained by a procedure for self-adapting the values of σ (Schwefel, 1981; analyzed in Beyer, 1996a). It is important to note that in the opposite case of a constant mutation strength, $\sigma = const.$, the (μ, λ)-ES would not be a function optimizer. That is, for $g \to \infty$ a residual distance R_∞ to the optimum remains $R_\infty = \sigma N / 2 c_{\mu,\lambda}$ (Beyer, 1993). Note, similar statements can be made for the mutation rate p_m in GAs (see e.g. Davis & Principe, 1993). Keeping p_m constant at a very small value is not always the best policy.

The governing principle of the ES progress theory can be extracted from Eq. 12:

Evolutionary Progress Principle (EPP)

EVOLUTIONARY PROGRESS	=	PROGRESS GAIN	−	PROGRESS LOSS

progress (positive or negative) is a result of the two opposite tendencies of progress gain and progress loss. The gain part of Eq. 12 is the $c_{\mu,\lambda}\sigma^*$ term which is obviously connected to the (μ, λ) truncation selection. Whereas the loss part $(\sigma^*)^2/2$ is independent of the selection process (this holds for the non-recombinant (μ, λ)-ES). The reason for this principle can be traced back to the effect of the (isotropic) mutation vector $\mathbf{z}$. A Careful analysis of the derivation leading to Eq. 12 reveals that it is always possible to decompose the mutation vector $\mathbf{z}$ into two parts: x in optimum direction $-\mathbf{e}_R$ and a perpendicular $\mathbf{h}$

$$\mathbf{z} = -x\mathbf{e}_R + \mathbf{h}, \tag{15}$$

depicted in Fig. 1. Thus, the progress gain is connected to x and the progress loss is due

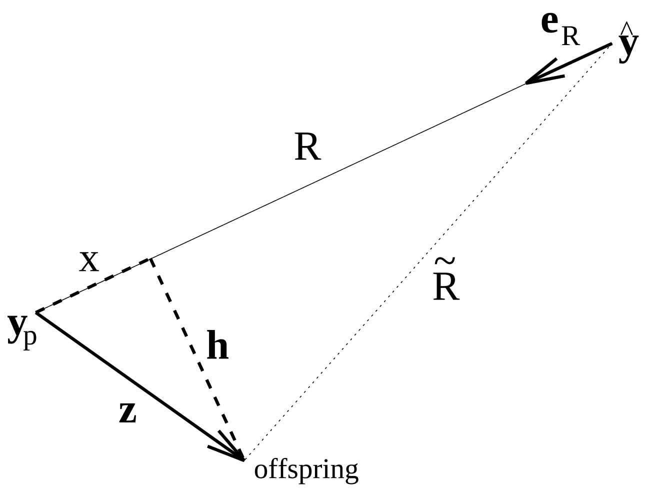

Fig. 1: Decomposition of the $\mathbf{z}$ mutation in the state space in a part x in optimum direction and a perpendicular part $\mathbf{h}$. $\mathbf{e}_R$ is the unity vector from $\hat{\mathbf{y}}$ to the parental state $\mathbf{y}_p$, R is the parental distance to the optimum $\hat{\mathbf{y}}$, and $\tilde{R}$ is the distance of the optimum to $\hat{\mathbf{y}}$.

to the length of $\mathbf{h}$. Each additional operator aims to decrease $||\mathbf{h}||$ and/or increase x.

It is important to realize that the EPP is not restricted to the spherical model. The decomposition Eq. 15 is always possible in real-valued parameter spaces and can be extended to integer and even binary parameter spaces.

The EPP is the governing principle for a general EA theory. It also builds the basis for the analysis and synthesis of "genetic" operators. This gives rise to a general operator design methodology which should be applied for each EA design:

1. Identify the gain and loss parts and the connections to the EA and its operators,
2. Identify the influence of the mutation strength,
3. Try to increase/decrease the gain/loss parts.

3.3.2 The $(\mu/\mu_I, \lambda)$-Theory

The asymptotic φ^* formula (Beyer, 1995b) is given as

$$\varphi^* = c_{\mu/\mu,\lambda}\, \sigma^* - \frac{(\sigma^*)^2}{2\mu}. \tag{16}$$

The progress coefficient $c_{\mu/\mu,\lambda}$ is of order

$$c_{\mu/\mu,\lambda} = \mathcal{O}\left(\ln(\lambda/\mu)\right) \tag{17}$$

and bounded by

$$0 \le c_{\mu/\mu,\lambda} \le c_{\mu,\lambda} \le c_{1,\lambda} \le \sqrt{2\ln\lambda}. \tag{18}$$

A comparison of the loss terms in Eq. 12 and Eq. 16 indicates that the intermediate recombination decreases the loss term $(\sigma^*)^2/2$ by a factor of $1/\mu$. The gain term is not increased because $c_{\mu/\mu,\lambda} \le c_{\mu,\lambda}$ holds. This important observation leads to the basic principle of intermediate recombination: given the decomposition of Eq. 15, the intermediate averaging in Eq. 7 is applied to the x-values[2] of the μ best offspring and to its $\mathbf{h}$-vectors as well. Selection acts mainly on the x-values, whereas the $\mathbf{h}$-vectors are (almost) selectively neutral. Moreover, the $\mathbf{h}$-vectors are statistically independent from each other. The averaging of these $\mathbf{h}$-vectors produces a recombinative $\mathbf{h}$-vector $\langle\mathbf{h}\rangle$ with an expected length quadrat $\overline{\|\langle\mathbf{h}\rangle\|^2}$ which is smaller by a factor of $1/\mu$ than the expected length quadrat of a single $\mathbf{h}$-vector

$$\overline{\langle\mathbf{h}\rangle^2} = \frac{1}{\mu}\overline{\mathbf{h}^2} \approx \frac{\sigma^2 N}{\mu}\,. \tag{19}$$

The $\mathbf{h}$-components of the mutation $\mathbf{z}$ may be considered as the "harmful part," because they increase the distance to the optimum (cf. Fig. 1), and a recombination producing

[2]Note, x does not coincide necessarily with one of the coordinate directions.

$\langle \mathbf{h} \rangle$ decreases this part. This process has been referred to as *genetic repair*[3] (GR) (Beyer, 1995b):

The GR-Principle

INTERMEDIATE RECOMBINATION	=	STATISTICAL ERROR CORRECTION	=	GENETIC REPAIR

Fig. 2 visualizes the GR-effect for a $(4/4_I, \lambda)$-ES in a two-dimensional intersection of the parameter space. The "iso-fitness hypersurfaces" are displayed by the curves $F(\mathbf{y}) = b$ and

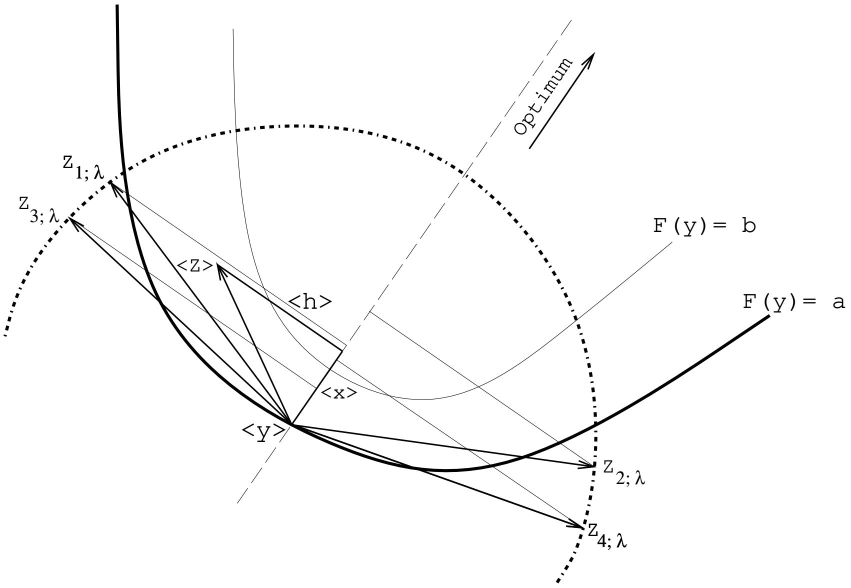

Fig. 2: Visualization of the genetic repair effect in intermediate recombinant strategies. The four best offspring out of the λ, generated from the parental state $\langle \mathbf{y} \rangle$ by the mutations $\mathbf{z}_{1;\lambda}, \dots \mathbf{z}_{4;\lambda}$, are displayed. The result of the recombination is $\langle \mathbf{z} \rangle$.

$F(\mathbf{y}) = a$, where the fitness value b is assumed to be fitter than a. From the parental state $\langle \mathbf{y} \rangle$ λ offspring are generated. The four best offspring, i.e. the mutations belonging to them, are displayed.[4] Due to the selection, $\mathbf{z}$-mutations are preferred which have a certain tendency into optimum direction. In the example of Fig. 2 all selected offspring are worse than the parent $\langle \mathbf{y} \rangle$. However, performing the recombination, i.e. the averaging of the

[3]The wording "genetic repair" was inspired by the DNA repair hypothesis of Bernstein et al. (1987) which is one of the current hypotheses on the function of recombination in biota (see Michod and Levin, 1988).

[4]Note, the $\mathbf{z}$-mutations can only schematically be displayed: The lengths of the $\mathbf{z}$'s are roughly $\sigma\sqrt{N}$. They lie nearly on a hypersphere (isotropic Gaussian mutations), however, they do not lie in the two-dimensional plane.

four best offspring produces a new parent which is fitter than all the offspring and the old parent. It is really important to realize that the effect of the intermediate recombination is some kind of *statistical error correction.*

It is evident that the building block hypothesis does not hold for intermediate recombination. The statement that recombination "combines good properties of the mates in order to form better offspring" is in error. In the above model, the "good properties" are simply the x-values of the μ best offspring. Since they are averaged, the result cannot be better than the maximal x of the best μ offspring. In other words, recombination dampens both "especially good" and "especially bad" properties of the mates.

Recombination extracts mainly the SIMILARITIES *from the selected offspring.*

System conditions of recombination. Recombinant EAs do not always necessarily possess a performance advantage in comparison with simple mutation-selection based EAs. The benefit of "sex" depends on certain conditions of the system EA & fitness landscape which can be subsumed as *system conditions of recombination*:

1. local "convexity" of the fitness landscape, or $P_s(\sigma) < \frac{1}{2}$ (see below),
2. statistical independence of the mutations,
3. sufficiently large mutation strength σ,
4. appropriate choice of ρ, μ, and λ; usually $\rho = \mu$ should be chosen.

The first point becomes quite clear, if one looks at Fig. 2. Recombination can be beneficial only if the local domain of success (domain of improved fitness w.r.t. the parent) is smaller than the locally worse part of the parameter space. Otherwise the performance could be improved by simply raising the mutations strength or the mutation rate (in the case of bit-strings). Since convexity is not very suitable for the binary space, one can use the success probability criterion $P_s < 1/2$ as an indicator for restricted success domains. P_s is simply the probability for improving the parental state by a single mutation of strength σ (or mutation rate p_m). As can be seen, this condition is not restricted to the spherical model.

The second point is closely connected to the EPP loss part reduction, expressed by Eq. 19. The reduction mechanism can only work if the harmful parts are statistically independent of each other. The GR-mechanism relies on the "genetic" differences of the mates. Note, this seems to be in analogy to biology where incest is relatively seldom observed. However, if inbreeding is enforced, one often obtains offspring with defects.

The third point is a very central one, because recombination in EAs is used with the intention to speed up the evolution process. The improved performance of the $(\mu/\mu_I, \lambda)$-ES is achieved for $\sigma^* = \mu c_{\mu/\mu,\lambda}$ in contrast to $\sigma^* = c_{\mu,\lambda}$ for the (μ, λ)-ES. That is, the optimal mutation strength in recombinant strategies is usually much larger than in non-recombinant strategies.

The fourth system condition states that multi-mixing ($\rho > 2$) should be preferred over the biological standard ($\rho = 2$). The best GR-effect can be obtained by choosing $\rho = \mu$, because it provides the smallest progress loss for a given (μ, λ) selection scheme.

3.3.3 The $(\mu/\mu_D, \lambda)$-Theory

The progress rate:

$$\varphi^* = \sqrt{\mu}\, c_{\mu/\mu,\lambda}\, \sigma^* - \frac{(\sigma^*)^2}{2}. \tag{20}$$

At first glance one might think that there is no GR (genetic repair) in dominant strategies, because there is no parental averaging in algorithm, Eq. 8. And it seems that the loss term in Eq. (20) is equal to that of the (μ, λ)-theory. However, looking at the *optimal* mutation strength $\hat{\sigma}^* = \sqrt{\mu} c_{\mu/\mu,\lambda}$ shows that the loss term of the optimally tuned $(\mu/\mu_D, \lambda)$-ES is equal to that of the optimally tuned $(\mu/\mu_I, \lambda)$-ES. Although both algorithms are seemingly different (as to the recombination operators), they are closely related from a probability theory perspective. Actually, Eq. 20 can be derived from the φ^*-formula for intermediate recombination, Eq. 16, by introduction of isotropic Gaussian *surrogate* mutations $\mathbf{S}$ with standard deviation σ_s

$$\sigma_s = \sqrt{\mu}\, \sigma. \tag{21}$$

The idea behind this model is that the effect of the dominant recombination can be emulated by mutations $\mathbf{S}$ (surrogates) which act on an imaginary center of mass parent $\langle \tilde{\mathbf{y}} \rangle_\mu$. This is possible, because each parent distribution (population) can be described by a probability density. The density can be expressed by its first moment – the center of mass – and the higher-order central moments. The application of dominant recombination and mutation changes these central moments, but the resulting distribution can still be interpreted as if it were generated from the center of mass of the parents by the surrogate mutation $\mathbf{S}$. A first approximation of this distribution can be obtained by neglecting the influence of selection on the shape of the N-dimensional surrogate mutation density. The examination of the single component standard deviation σ_s of $\mathbf{S}$ suffices. (This holds exactly for the asymptotic $N \to \infty$ case.)

Consider a population (comprising μ elements) from a single (vector) component s of $\mathbf{S}$. Let this population be subject to an evolutionary process consisting of recombination and a "physical" mutation with strength σ iteratively applied, but without selection. The time evolution of the σ_s can be calculated (Beyer, 1996b)

$$\sigma_s^{(g+1)} = \sigma \sqrt{\frac{1 - \left(\frac{\mu-1}{\mu}\right)^{g+1}}{1 - \left(\frac{\mu-1}{\mu}\right)} + \left(\frac{\mu - 1}{\mu}\right)^{g+1} \left(\frac{\sigma_s^{(0)}}{\sigma}\right)^2}. \tag{22}$$

There are two steady state scenarios for $g \to \infty$.

First, when $\sigma > 0$, the standard deviation of the surrogates becomes $\sigma_s \to \sqrt{\mu}\sigma$ (cf. Eq. 21) independent of the initial parental $\sigma_s^{(0)}$. The transient time for this process is proportional to μ. One obtains a stable parent distribution which is Gaussian shaped and "crowded" around an imaginary center of mass parent. One is tempted to compare this with the distribution of a species in biology. The imaginary center of mass parent resembles the wild-type.

Second $\sigma = 0$, but $\sigma_s^{(0)} > 0$. One can easily derive from Eq. 22 that σ_s approaches zero. That is, the population shrinks with $g \to \infty$. In other words, for $g \to \infty$ one observes a fixation of the center of mass parent. The actual fixed point is at random. This behavior is very similar to the phenomenon of *random genetic drift* in biota (cf. e.g. Christiansen and Feldman, 1986, p. 70). Actually the fixation time is roughly 2μ, i.e., it is proportional to the reproductive part of the offspring population, known in population genetics as the effective population size. This result is in agreement with the results of the population genetics (Christiansen and Feldman, 1986, p. 71). It is important to realize that the effect of random genetic drift is not equivalent to a *random walk*. The latter can be obtained by switching off the recombination with the result of a simple diffusion process which obeys a $\sqrt{g}$-law. That is, the standard deviation σ_s of the population becomes $\sigma_s^{(g)} = \sigma\sqrt{g}$ and the population disperses in the state space.

As can be seen, examining the effects of mutation and recombination in isolation is insufficient. There is a synergism that arises from the interplay of the different operators resulting in a population which resembles a species. The "species" is produced by recombination of the mutated individuals. This is regarded as a basic principle of dominant recombination and has been coined *mutation-induced speciation by recombination* (MISR)

MISR-Principle

DOMINANT RECOMBINATION	&	PHYSICAL MUTATIONS	$\Rightarrow$	"SPECIES"

Genetic Repair in Dominant Recombinant Strategies. The effect of the MISR-principle is the transformation of the physical mutations into larger surrogate mutations or "genetic" variation. If this would be the only effect of the dominant recombination, then it could be easily attained by a non-recombinant (μ, λ)-strategy with a larger physical mutation strength. The GR-effect in the intermediate recombinant strategies, however, is based upon the explicit calculation of the center of mass. Though there is not an explicit GR-procedure in $(\mu/\mu_D, \lambda)$-ESs, there is still an implicit GR-mechanism. This becomes clear if one interprets the dominant recombination as a sampling process that estimates the parental distribution. Thus, the center of mass – as the first moment of the distribution – is implicitly estimated. Fig. 3 serves to clarify this estimation process. One can interpret this estimation process as an statistical extraction of similarities from the selected offspring. Therefore, the effect of the dominant recombination can be summarized as

GR on Dominant Recombination

Dominant Recombination	=	Statistical Extraction of Similarities	=	Approximate, Statistical Error Correction (GR)

4 The Correspondence between Principles in Evolution Strategies and Genetic Algorithms

The three principles outlined in section 3 may be described as:

1. evolutionary progress principle (EPP),
2. mutation-induced speciation by recombination (MISR),
3. genetic repair (GR) hypothesis.

Consider the connections between this principles in evolution strategies and genetic algorithms.

4.1 The EPP in GAs

The evolutionary progress principle (EPP) postulates that progress is the result of the two opposite tendencies progress gain and progress loss. Progress gain is mainly connected to the selective forces in the algorithm, whereas the progress loss is due to the harmful parts of the (unbiased!) mutations. Though this principle seems to be very reasonable and has been proved for each ES analyzed up to now, there are only few results in GA theory that could be used to support the EPP hypothesis. The main reason for this is found in the way of measuring performance in GAs. Usually only global performance measures are used

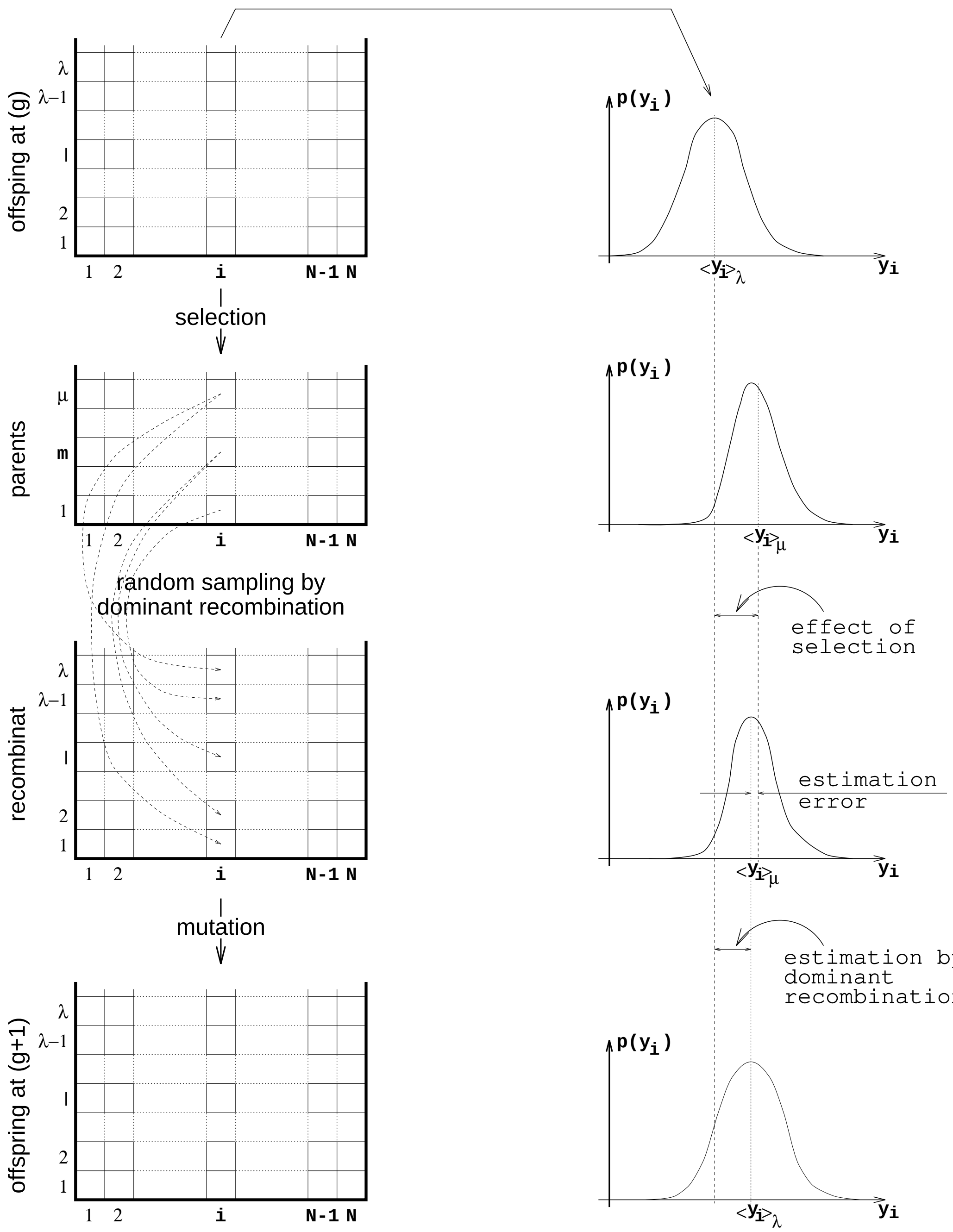

Fig. 3: The implicit GR-mechanism (statistical center-of-mass-determination) of the dominant recombination for the ith coordinate. The statistical parameters of the parent distribution are estimated by the "dominant sampling". Thus, the *similarities* of the parents are transferred to the offspring.

to compare the different algorithms. For example, the number of generations needed to reach a certain vicinity of the optimum is investigated. However, such measures say only little about how the EA approaches the optimum in the time domain. For this reason the evolutionary dynamics should be investigated which is constituted by the generational change of the population toward the optimum. Therefore, if one is interested in determining how fast an EA approaches the optimum, and if one wants to compare different algorithms, then the local performance should be investigated. There is obviously a lack of GA-investigations in this direction.

The only exception is for the `OneMax`-function $F(\mathbf{b})$, Eq. 2, concerning the simple mutation-selection-algorithm with $(1, \lambda)$ or $(1 + \lambda)$ truncation selection. With a technique offered in Beyer (1994a) one can derive progress rate formula that are exact in the asymptotic $\ell \to \infty$ limit (ℓ string length)

$$\varphi_{1,\lambda}(p_m) = c_{1,\lambda} \sqrt{p_m \, \ell} \, \sqrt{1 - p_m} \; - \; p_m \, \ell \left(2 \frac{F(\mathbf{b})}{\ell} - 1 \right), \tag{23}$$

Where $F(\mathbf{b})$ is given by Eq. 2 (i.e., the number of correct bits at generation g), p_m is the bit-flipping probability (mutation rate), and φ measures the generational change to the optimum using the Hamming distance. As one can clearly see, for $F > \ell/2$ the second term in Eq. 23 is negative and can be identified with the progress loss, whereas the first term is due to the truncation selection expressing the gain part of the EPP. Furthermore, φ is a function of the mutation rate p_m. Equation 23 can be used to determine the optimal mutation rate $\hat{p}_m$ leading to a quadratic equation with the approximate solution ($F > \ell/2$, $\ell \to \infty$, see Beyer, 1996b)

$$\hat{p}_m \sim \frac{c_{1,\lambda}^2}{4(2F/\ell - 1)^2} \, \frac{1}{\ell}. \tag{24}$$

Note, for a given relative distance-to-optimum value F/ℓ, the optimal mutation rate $\hat{p}_m$ is inversely proportional to the string length ℓ. This is in agreement with the commonly used mutation rate in standard GAs and also with the plausibility arguments of Mühlenbein (1992). However, the most important message from Eq. 24 is that the optimal mutation rate depends on the actual F-state of the parent.

4.2 The MISR-Principle

The *mutation-induced speciation by recombination principle* (MISR) applied to the binary GA postulates for each bit position a larger but restricted population variance σ_s^2 than the single bit-flipping variance generated with the mutation rate p_m. This is because the physical mutations (bit-flips) are transformed by the dominant recombination (random sampling) into "genetic" variation which looks like a "species-like" population. The variance of a single bit-flip is $\sigma^2[b] = p_m \, (1 - p_m)$. The analytical determination of the (averaged) population variance σ_s^2 has not been performed up until now. Note, Eq. 21 cannot be used directly, because it holds for continuous variates with zero mean mutations. There is not an upper limit in Eq. 21, whereas the variance of a bit-population is always bounded,

$\sigma_s^2 \leq 1/4$. However, the effect of the MISR-principle can be easily simulated and verified by iterated application of the operators mutation and recombination on a bit-population. Since selection is switched off, the center of mass (defined as the bit-count divided by the population size μ) performs a random walk, whereas the averaged (over the generations) population variance proves to be a function of p_m, monotonously increasing with μ. A comparison of the experiments (not presented here) with the predictions of Eq. 21 shows that it may serve as a rough estimate, if p_m is sufficiently small.

Apart from $p_m > 0$, there is the interesting $p_m = 0$, $\sigma_s^{(0)} > 0$ case, that produces the random genetic drift, investigated in GA-theory by Asoh and Mühlenbein (1994). As in the case of the continuous model, Eq. 22, they obtained a fixation time proportional to the population size indicating a further connection between ESs and GAs.

4.3 The GR-Hypothesis for GAs

Since there is no direct and totally satisfactory way (but, see below) to transfer the center of mass generation from the real-valued intermediate recombination of the ES into the GA, one has to consider the dominant recombination which is in relation to the uniform crossover. For the dominant recombination the GR is implicitly realized by statistical similarity extraction from the selected offspring. As to the GAs this conjecture has neither theoretical nor direct empirical proofs, however, some indirect evidences exist.

Unlike the ES theory, where GR is well understood as a principle that works beneficially in curved, i.e. nonlinear, fitness landscapes, the main characteristic of GAs is not yet extracted. What is the analogue to "curved" in GAs?

Curvature in ESs can be measured by the success probability P_s.[5] P_s is a function of the mutation strength σ. Exploiting the *ES-GA correspondence principle:*

$$\sigma \quad \longleftrightarrow \quad p_m \tag{25}$$

that postulates the analogue of mutation strength σ and the mutation rate p_m, then "curvature" in GAs on binary fitness landscapes could be expressed by $P_s(p_m) > 1/2$ (concave) and $P_s(p_m) < 1/2$ (convex), respectively. GR should be beneficial in fitness landscapes with $P_s(p_m) < 1/2$.

For the `OneMax`-function $F(\mathbf{b})$, Eq. 2, one can determine the success probability $P_s(p_m)$ in the asymptotic limit case (Beyer, 1996b)

$$P_s(p_m) = 1 - \Phi\left(\sqrt{p_m\,\ell}\;\frac{2F(\mathbf{b})/\ell\,-\,1}{\sqrt{1-p_m}}\right), \tag{26}$$

[5] P_s is the probability of producing a successful (fitter) offspring from the parental state by a single trial.

where $\Phi(x)$ is the cumulative distribution function of the standard normal variate. As one can see

$$F(\mathbf{b}) > \frac{\ell}{2} \quad \Rightarrow \quad P_s(p_m) < \frac{1}{2} \quad \Leftrightarrow \quad \text{“convex”}. \tag{27}$$

The notion “convex” is to be taken *cum grano salis*, however, $F(\mathbf{b}) > \ell/2$ seems to define the `OneMax`-domain for which recombination could be beneficial. A similar statement has been given by Mühlenbein and Schlierkamp-Voosen (1994): recombination (for `OneMax`) should be useful, if most of the bits are correct. This is in full agreement with the GR-hypothesis: Since the incorrect bit positions (the “`0`”) are not correlated and the probability of picking up a “`0`” is smaller than 1/2 (because $F(\mathbf{b}) > \ell/2$) dominant recombination will sample a `1`-position with a higher probability than a `0`-position. This can be interpreted as statistical extraction of similarities.

Provided that the GR-hypothesis in $P_s < 1/2$ landscapes is correct, then there remain two open questions concerning points 3 and 4 of the “system conditions of recombination” from above.

First, the benefit of recombination can only fully be used if the mutation rate p_m is optimally tuned. Second, multi-mixing should be used in order to improve the GR-effect (Beyer, 1994b), however, it is an open problem how to perform the selection (how to choose μ and λ, or even other selection schemes like steady state, tournament, or proportional selection).

Using multi-mixing in GAs has been also reinforced in a number of papers by Eiben et al. (1994, 1995, 1996) which may serve as a clue supporting the GR-hypothesis. They investigated simple GAs (proportionate selection) with multi-parent recombination, the so-called “gene scanning.” Among the different variants there have been two of special interest. The first variant is the “uniform gene scanning” which is similar to the dominant recombination described in section 3. The second variant is the “occurrence-based scanning,” a straightforward implementation of the intermediate recombination on binary strings. However, since the averaging of bits would produce fractional bits, the analogy breaks down.[6] Therefore, the intermediate bit value is determined by majority decision (occurrence-based).[7]

The results obtained by Eiben et al. (1994) are promising: “The experiments show that 2-parent recombination is inferior on the classical De Jong functions. For the other problems the results are not conclusive, in some cases 2 parents are optimal, while in some others more parents are better.”

Although this work provides indirect support for the GR-hypothesis it cannot be interpreted as empirical evidence because global performance measures have been used. Further

[6]From the viewpoint of ES theory, this indicates that the binary alphabet should be exchanged by higher cardinality alphabets (if possible). GR could work much better.

[7]Note, this technique can produce large round off errors diminishing the effect of GR.

work is to be done concerning the *local* speedup (local performance measures!) of the multi-mixing GAs.

5 Concluding Remarks

In this paper basic EA principles derived from the ES theory have been used to discuss an alternative hypothesis on how GAs work. It was argued that the building block hypothesis does not explain the evolutionary mechanisms, i.e. the *cause* that makes a GA a function optimizer or not. The *observed* building block evolution is rather an *effect* of each function optimizer which successively approaches the optimum.

The main goal of this paper was to shed light on the process of mutation/recombination. Whereas, the aspects of selection (see e.g. Mühlenbein and Schlierkamp-Voosen, 1994) and the prerequisite for evolvability (Rechenberg, 1994) which is closely connected to the question of the right (problem specific) coding have been omitted here. They may be regarded as additional basic principles/conditions of similar importance.

Let us give a short summary of what the evolutionary progress principle (EPP), the mutation-induced speciation by recombination principle (MISR), and the genetic repair (GR) hypothesis mean as to GAs.

The governing principle is the EPP. Each designer of a GA (or generally EA) should ask for the processes that provide progress gain and progress loss. The analysis of schema disruption/production, however, is not equivalent to progress loss/gain. Therefore, in general, the results of schema analysis will not be as significant as those obtained by EPP (though it may be more difficult to extract EPP).

Once the EPP is accepted, GR aims at the decrease of the loss part by *statistical* error correction. Thus, it is possible to increase the mutation rate p_m with the effect of a higher progress gain. Once again, mutations are not directed and without bias: A larger mutation rate p_m increases *both* the fitness increasing components *and* the harmful components of a mutation. Selection extracts those individuals produced by the mutations with the larger fitness increasing components (on average), and the recombinative GR-mechanism diminishes the influence of their harmful parts. Thus, it is possible to explain the benefits of recombination without reference to the building block hypothesis.

As already pointed out, the magnitude of the mutations is a crucial parameter for the EA performance. This holds for the spreading of the population as well, because the whole population can be interpreted as if it were produced by surrogate mutations from an average parent (center of mass parent).

GR-performing populations are very similar to species: the individuals are crowded around the average individual. The MISR-hypothesis explains why the small physical mutations are transformed into larger (surrogate) mutations at the population level by means of the recombination/crossover operator. However, these mutations are still restricted which is a prerequisite for GR.

At least for the multirecombinant ES, the "species" produced by MISR performs GR well. It is an open question whether or not this holds for GAs, too, especially if the standard recombination operators one-point/two-point crossover are considered. It might well be possible that the main effect of these standard crossover operators can be attributed to the MISR principle, whereas GR is only poorly performed. If this were the case, then there would be no need for crossover at all, because the population spreading effect of the MISR principle can be emulated by an enlarged mutation rate.

Further research will be needed to support the hypotheses proposed for the binary GAs. This should be done both in theory and by computer experiments. The main emphasis is on the genetic repair hypothesis, because it is the most delicate part explaining the evolution accelerating behavior of recombination. Such a research program should include the investigation of the representation problem (i.e., the coding used) of the GA. It may be possible that, e.g., the binary representation of real-valued optimization problems will be not the best choice with respect to the effect of GR due to the high granularity in the (most) significant bits.

Acknowledgments

I'm grateful to David B. Fogel for discussions concerning the role of recombination with respect to evolutionary programming (EP) and its philosophical background. He has greatly influenced the final version of this article and pointed me to the "speciation" notion to be used in MISR.
This work was funded by DFG grant Be 1578/1-1, /1-2, and /1-3.

References

[1] L. Altenberg, 1995. The Schema Theorem and Price's Theorem. In M. D. Vose and L. D. Whitley, editors, *Foundations of Genetic Algorithms, 3*, pages 23–49. Morgan Kaufmann, San Mateo, CA.

[2] H. Asoh and H. Mühlenbein, 1994. On the Mean Convergence Time of Evolutionary Algorithms without Selection and Mutation. In Y. Davidor, R. Männer, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature, 3*, pages 88–97, Heidelberg, Springer.

[3] H.-G. Beyer, 1993. Toward a Theory of Evolution Strategies: Some Asymptotical Results from the $(1,^{+}\lambda)$-Theory. *Evolutionary Computation*, 1(2):165–188.

[4] H.-G. Beyer, 1994a. Towards a Theory of 'Evolution Strategies': Progress Rates and Quality Gain for $(1,^+\lambda)$-Strategies on (Nearly) Arbitrary Fitness Functions. In Y. Davidor, R. Männer, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature, 3*, pages 58–67, Heidelberg, Springer-Verlag.

[5] H.-G. Beyer, 1994b. Towards a Theory of 'Evolution Strategies': Results from the N-dependent (μ, λ) and the Multi-Recombinant $(\mu/\mu, \lambda)$ Theory. Technical Report SYS–5/94, Department of Computer Science, University of Dortmund.

[6] H.-G. Beyer, 1995a. Toward a Theory of Evolution Strategies: The (μ, λ)-Theory. *Evolutionary Computation*, 2(4):381–407.

[7] H.-G. Beyer, 1995b. Toward a Theory of Evolution Strategies: On the Benefit of Sex - the $(\mu/\mu, \lambda)$-Theory. *Evolutionary Computation*, 3(1):81–110.

[8] H.-G. Beyer, 1995c. How GAs do *NOT* Work - Understanding GAs without Schemata and Building Blocks. Technical Report SYS–2/95, Department of Computer Science, University of Dortmund.

[9] H.-G. Beyer, 1996a. Towards a Theory of Evolution Strategies: Self-Adaptation. *Evolutionary Computation*, 3(3):311–347.

[10] H.-G. Beyer, 1996b. *Zur Analyse der Evolutionsstrategien.* Habilitation, University of Dortmund. To be published.

[11] H. Bernstein, F. A. Hopf, and R. E. Michod, 1987. The Molecular Basis of the Evolution of Sex. *Advances in Genetics*, 24:323–370.

[12] F. B. Christiansen and M. W. Feldman, 1986. *Population Genetics.* Blackwell Scientific Publications, Palo Alto, CA.

[13] T. E. Davis and J. C. Principe, 1993. A Markov Chain Framework for the Simple Genetic Algorithm. *Evolutionary Computation*, 1(3):269–288.

[14] K. A. De Jong, 1992. Are genetic algorithms function optimizers? In R. Männer and B. Manderick, editors, *Parallel Problem Solving from Nature, 2*, pages 3–13. North Holland, Amsterdam.

[15] K. A. De Jong, W. M. Spears, and D. F. Gordon, 1995. Using Markov Chains to Analyze GAFOs. In M. D. Vose and L. D. Whitley, editors, *Foundations of Genetic Algorithms, 3*, pages 115–137. Morgan Kaufmann, San Mateo, CA.

[16] A. E. Eiben, C. H. M. van Kemenade, and J. N. Kok, 1995. Orgy in the Computer: Multi-Parent Reproduction in Genetic Algorithms. In F. Morana, A. Moreno, J. J. Merelo, and P. Chacon, editors, *Advances in Artificial Life. Third ECAL Proceedings*, pages 934–945, Berlin, Springer-Verlag.

[17] A. E. Eiben, P.-E. Raué, and Z. Ruttkay, 1994. Genetic algorithms with multi-parent recombination. In Y. Davidor, R. Männer, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature, 3*, pages 78–87, Heidelberg, Springer-Verlag.

[18] A. E. Eiben and C. A. Schippers, 1996. Multi-parent's niche: n-ary crossover on NK-landscapes. In *Parallel Problem Solving from Nature, 4*, Heidelberg, Springer-Verlag.

[19] D. B. Fogel and J. W. Atmar, 1990. Comparing genetic operators with Gaussian mutations in simulated evolutionary processes using linear systems. *Biological Cybernetics*, 63:111–114.

[20] D. B. Fogel, 1992. *Evolving Artificial Intelligence*. PhD thesis, University of California, San Diego.

[21] D. B. Fogel and L. C. Stayton, 1994. On the effectiveness of crossover in simulated evolutionary optimization. *BioSystems*, 32:171–182.

[22] D. E. Goldberg, 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison Wesley, Reading, MA.

[23] J. J. Grefenstette, 1991. Conditions for Implicit Parallelism. In G. J. E. Rawlins, editor, *Foundations of Genetic Algorithms, 1*, pages 252–261. Morgan Kaufmann, San Mateo, CA.

[24] J. J. Grefenstette, 1993. Deception considered harmful. In L. D. Whitley, editor, *Foundations of Genetic Algorithms, 2*, pages 75–91. Morgan Kaufmann, San Mateo, CA.

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

[26] J. H. Holland, 1992. *Adaptation in Natural and Artificial Systems.* MIT Press, Cambridge, MA. 2nd. edition.

[27] J. H. Holland, 1995. *Hidden Order: How Adaptation Builds Complexity.* Addison-Wesley, Reading, MA.

[28] R. E. Michod and B. R. Levin, editors, 1988. *The Evolution of Sex: an Examination of Current Ideas.* Sinauer Associates Inc., Sunderland, MA.

[29] L. Margulis and D. Sagan, 1988. The Cannibalistic Legacy of Primordial Androgynes. In R. Belling and G. Stevens, editors, *Nobel Conference XXIII (1987): The Evolution of Sex*, pages 23–40. Harper and Row, San Francisco.

[30] H. Mühlenbein, 1992. How Genetic Algorithms Really Work I: Mutation and Hillclimbing. In R. Männer and B. Manderick, editors, *Parallel Problem Solving from Nature, 2*, pages 15–25. North Holland, Amsterdam.

[31] H. Mühlenbein and D. Schlierkamp-Voosen, 1994. The science of breeding and its application to the breeder genetic algorithm BGA. *Evolutionary Computation*, 1(4):335–360.

[32] I. Rechenberg, 1994. *Evolutionsstrategie '94*. Frommann–Holzboog Verlag, Stuttgart.

[33] G. Rudolph, 1994. Convergence properties of canonical genetic algorithms. *IEEE Transaction on Neural Networks*, 5(1):96–101.

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

[35] H.-P. Schwefel, 1995. *Evolution and Optimum Seeking.* Wiley, New York, NY.

[36] H.-P. Schwefel and G. Rudolph, 1995. Contemporary Evolution Strategies. In F. Morana, A. Moreno, J. J. Merelo, and P. Chacon, editors, *Advances in Artificial Life. Third ECAL Proceedings*, pages 893–907, Berlin, Springer-Verlag.

[37] G. Syswerda, 1989. Uniform Crossover in Genetic Algorithms. In J. D. Schaffer, editor, *Proc. 3rd Int'l Conf. on Genetic Algorithms*, pages 2–9, San Mateo, CA, Morgan Kaufmann.

[38] M. D. Vose and G. E. Liepins, 1991. Punctuated equilibria in genetic search. *Complex Systems*, 5(1):31–44.

[39] M. D. Vose and A. H. Wright, 1995. Simple genetic algorithms with linear fitness. *Evolutionary Computation*, 2(4):347–368.

[40] L. D. Whitley, 1991. Fundamental Principles of Deception in Genetic Search. In G. J. E. Rawlins, editor, *Foundations of Genetic Algorithms, 1*, pages 221–241. Morgan Kaufmann, San Mateo, CA.