On the Analysis of Self-Adaptive Evolutionary Algorithms

Hans-Georg Beyer
Systems Analysis Group
Department of Computer Science XI
University of Dortmund
44221 Dortmund, Germany
beyer@zappa.cs.uni-dortmund.de

Kalyanmoy Deb
Kanpur Genetic Algorithms Laboratory (KanGAL)
Department of Mechanical Engineering
Indian Institute of Technology Kanpur
Kanpur, PIN 208 016, India
deb@iitk.ac.in

Technical Report No. CI-69/99
May 1999
Department of Computer Science/XI
University of Dortmund
44221 Dortmund, Germany

Abstract

Due to the flexibility in adapting to different fitness landscapes, self-adaptive evolutionary algorithms (SA-EAs) have been gaining popularity in the recent past. In this paper, we postulate the properties that SA-EA operators should have for successful applications. Specifically, population mean and variance of a number of SA-EA operators, such as various real-parameter crossover operators and self-adaptive evolution strategies, are calculated for this purpose. In every case, simulation results are shown to verify the theoretical calculations. The postulations and population variance calculations explain why self-adaptive GAs and ESs have shown similar performance in the past and also suggest appropriate strategy parameter values which must be chosen while applying and comparing different SA-EAs.

Keywords

Self-adaptation, Population mean, Population variance, Evolution strategies, Genetic algorithms, Simulated binary crossover, Blend crossover operator, Fuzzy recombination operator

I. Introduction

Among the evolutionary algorithms (EAs), explicit studies on self-adaptation have been made only with evolution strategies (ESs) [1] and evolutionary programming (EP) [2]. Self-adaptive evolutionary algorithms (SA-EAs) use operators which are adaptive in some sense—either the strategy parameters controlling the extent of search are evolved [1] or are updated based on statistics gathered from past generations [3]. Since the operators are flexible, they have been shown to adapt to a variety of fitness landscapes better than EAs which do not use specific self-adaptive operators. For example, when a population is located in the basin of attraction of an optimum, SA-EAs have been shown to converge to the optimum exponentially quickly. They have also been shown to adapt to changing fitness landscapes, that is, self-adaptive EAs have been found to quickly get out of the current optimum and proceed towards the new optimum, as and when the fitness landscape changes. If initialized away from the optimum, SA-EAs have also shown to diversify its population exponentially faster to reach near the optimum.

Recently, the self-adaptive feature of a number of real-parameter GAs is also demonstrated on similar fitness landscapes [4], [5], [6]. Unlike evolving or calculating the strategy parameters, these GAs use crossover operators which create children solutions proportional to the difference in parent solutions. Since children population is controlled indirectly by the spread of parent population, convergence, divergence, or adapting to changing fitness landscapes are possible to achieve. These crossover operators create one or two children solutions according to a probability distribution over two or more parent solutions. The most popular approach has been to use a uniform probability distribution (Blend crossover (BLX) suggested in [7]) around a region bracketing the parent solutions. There exists at least three different approaches where a non-uniform probability distribution has been suggested. Of them, the simulated binary crossover (SBX) uses a bimodal probability distribution with its mode at the parent solutions [8]. This is similar to fuzzy recombination operator (FR) which also uses a bimodal distribution, but the distribution around each parent is always triangular [6]. Ono and Kobayashi [9] suggested a unimodal normally distributed crossover (UNDX) operator which uses an ellipsoidal probability distribution around three parents to create a child solution.

The authors [10] and Kita [5] have shown earlier that SBX and UNDX operators can provide real-parameter GAs self-adaptive power similar to that of self-adaptive ESs on a number of problems. The former study by the authors has drawn a connection between the working principle of real-parameter GAs with SBX and isotropic self-adaptive ESs. GAs with SBX uses a probability distribution which is controlled adaptively in proportion to the difference in parent solutions, whereas a self-adaptive ES uses a probability distribution of similar nature but is controlled by explicit mutation strength parameters which are also evolved along with decision variables. Although this connection was not made any more rigorous, it was felt that there exists a much deeper connection between the working of real-parameter GAs with SBX and other similar crossover operators and self-adaptive ESs. In this paper, we attempt to make the connection between their workings more rigorous by calculating the mean and variance of resulting children population from that of the parent population. We support our arguments by performing simulations on a number of test problems.

In the remainder of the paper, we present some postulates on the behavior of self-adaptive EAs on a number of fitness landscapes. Thereafter, we compute the properties of three crossover operators (BLX, SBX, and FR) and self-adaptive ESs. Theoretical calculations of their behaviors are verified with simulation results. Although the implementation of these EAs are different, the analysis shows that underlying working principle of them are very similar, if appropriate characteristic parameters are chosen for each operator.

II. Some Postulates on the Behavior of Self-Adaptive Evolutionary Algorithms

In the discussions here, we consider an evolutionary algorithm having two major operators—reproduction and variation (in the usual EAs, this refers to recombination and mutation operators). In our argument, the sequence of their operations is not important. Under the reproduction operation, a population of μ solutions is expected to loose some not-so-good solutions, only to make duplicates of some good solutions. Although it may not be true in general for any arbitrary fitness landscape and for any reproduction operator, this process of deleting some solutions and making duplicates of some other solutions have an effect of reducing the variance of population. That is, the variance of population in the search space after the reproduction operation is, in most situations, smaller than that before the reproduction operation. This is particularly true for unimodal or linear fitness landscapes. For multi-modal fitness landscapes, this may not be true always, particularly when the solutions near the population mean are bad solutions. Although the variance may increase in these cases after the reproduction operation, it is important to note that the maximum spread[1] of the population does not increase due to the reproduction operation. This is because no new solution is created in the reproduction operation.

Based on this observation on reproduction operation, we now postulate some desired properties of variation operators.

A. Postulates on population mean

The purpose of a variation operator is to use the parent population formed after the reproduction operation to create a children population. Although it is not absolutely necessary, most GA variation operators do not use any fitness information, instead partial information between a pair of parents are exchanged. It is argued in the literature [11] that the purpose of the reproduction operation is to *exploit* the search space by emphasizing *good* solutions and the purpose of the variation operator is to *explore* the search space. If both operators use fitness information, there may be too much emphasis of exploitation and resulting GAs may have properties of premature convergence. Since an ideal variation operator may not use the fitness information, we may postulate the following:

Proposition 1: Under a variation operator, the expected population mean should remain unchanged.

Standard crossover operators used in binary-coded GAs preserve this property. We shall prove it for the single-point crossover operator on a pair of parent strings representing a discrete search space. Let us say a and b are two parent Boolean vectors in $\{0,1\}^\ell$. Let us also assume that the crossing over takes place at the k-th site. The mean of the parents is given by:

$$\bar{x} = \sum_{i=0}^{\ell-1} 2^i \left(a_i + b_i\right)/2. \tag{1}$$

After crossover, the decoded value of the children solutions y_1 and y_2 are as follows:

$$y_1 = \sum_{i=0}^{k} 2^i a_i + \sum_{i=k+1}^{\ell-1} 2^i b_i, \tag{2}$$

[1] The maximum distance between any two solutions.

$$y_2 = \sum_{i=0}^{k} 2^i b_i + \sum_{i=k+1}^{\ell-1} 2^i a_i. \quad (3)$$

The mean of children solutions is $\bar{y} = 0.5(y_1 + y_2)$. Substituting y_1 and y_2 terms from above two equations, one observes that $\bar{y} = \bar{x}$. Thus, the mean of a pair of parent solutions after the single-point crossover is the same as the mean of resulting pair of children solutions. It is also obvious that the above property is true for multi-point crossover and uniform crossover operators. Avoiding the sampling issue of a finite population, it can also be argued that the expected population mean of a parent population is the same as the expected population mean of the children population.

Standard mutation operators in EAs do not use fitness information, instead use a zero-mean probability distribution to create a mutated child. Thus, the expected population mean is not expected to change under a standard mutation operator.

The proposition also makes sense intuitively. Since the variation operator, in general, does not use fitness information, there is no reason for it to shift the parent population mean in any direction in the search space. It is the task of the reproduction operator to lead the population in an appropriate direction in the search space. The best the variation operator can do is to keep the population mean invariant, but increase or reduce the spread (or variance) of the resulting population. In the following subsection, we describe postulates for increasing or decreasing population variance under a variation operator.

B. Postulates on population variance

It is argued earlier that depending on the fitness landscape and the position of the parent population, the population variance may decrease or increase after the reproduction operation. Thus, to avoid any premature convergence or stagnation, the population after variation operation must adjust its variance accordingly, so as to keep the overall population variance to be of reasonable value from generation to generation. Particularly, if the reproduction operator reduces the variance of population, the variation operator must increase the variance of population, and vice versa. In the following, we consider postulates for a few fitness landscapes.

1. Flat fitness landscapes:

Proposition 2: The expected population variance $\sigma^2[y]$ should exponentially increase[2] with generation number g.

For the definition of $\sigma^2[y]$, see Section III-B, Eq. (22). Although not totally obvious, the intention here is leave the region of constancy as fast as possible, simply because there is no selection preference for any part of the search space. We would like to mention that this postulation is different from that in Ostermeier [12, p. 18], who suggests a zero drift in a similar situation.

2. Linear fitness landscapes (hyperplane):

Proposition 3: The expected population variance $\sigma^2[y]$ should exponentially increase with g.

Here, the reason is clear. Linear fitness landscapes divide the search space into two subspaces separated by a hyperplane. Considering isotropic (Gaussian) mutations, on average each second mutation (starting from a parental point) yields an improvement independent of the mutation strength used. However, the larger the mutations the larger the expected improvement. Therefore, the mutation strength and the population variance, respectively, should increase over the generations. The same argument can be made for a recombination operator. Instead of the above postulate on population variance, one can postulate that the distance traveled into the optimum direction by the parents should increase exponentially in order to reach near the optimum quickly.

3. Unimodal fitness landscapes:

If the parent population brackets the optimum, a controlled reduction in population variance by the variation operator is desired. Whereas, if the parent population lies in one side of the optimum, the situation is similar to that in the linear fitness landscape and the variation operator should increase the population variance in order to reach near the optimum quickly. In an arbitrary problem, the latter is more common and therefore it is better to use a variation operator which increases the population variance. It may be worthwhile to mention here that a popular bracketing algorithm in the classical optimization explicitly uses a strategy of approaching exponentially faster towards the optimum [13] in unimodal problems.

4. Multi-modal fitness landscapes:

The increase or decrease of population variance should largely depend on the fitness landscape and placement of the parent population. We argue that even in these cases, a generic strategy would be to use

[2] Exponential increase may regarded as a strong form of SA. As a weak variant one might postulate expected monotonous increase.

a variation operator which increases the population variance and then introduce an adequate selection pressure by the reproduction operator to exploit the children population. We believe that this strategy has a better global perspective than using variation operators which would reduce the population variance.
5. Moving target problems (time dependent optimum) and rapidly changing fitness landscapes:
In both cases it may be of importance to *increase* the mutation strength/population variance after a period of shrinking. Therefore, the algorithms should have the ability to increase the variance. A simple way to ensure such a behavior is to implement a certain bias toward an increase of the population variance.

It is important to note that the above postulations should locally hold for each coordinate direction independently. That is, although the overall fitness function may not be flat with respect to all coordinate directions but is selectively neutral with respect to some coordinate directions, the postulation for the flat fitness landscape case should hold for those coordinate directions:

$$\forall\, \delta \in [-\delta_l, \delta_u]: \quad f(y_1, \ldots, y_k, \ldots, y_N) = f(y_1, \ldots, y_k + \delta, \ldots, y_N) \tag{4}$$

or which are linearly separable (the linear fitness landscape case):

$$\forall\, \kappa \in [-\kappa_l, \kappa_u]: \quad f(y_1, \ldots, \kappa y_k, \ldots, y_N) = f(y_1, \ldots, y_k, \ldots, y_N) + \kappa f_k. \tag{5}$$

The postulations should also hold for each arbitrary direction $\mathbf{v}$ ($||\mathbf{v}|| = 1$) in the search space. For example, the postulation for the flat fitness landscape should locally hold (with respect to a given parental state $\mathbf{y}$) for each direction $\mathbf{v}$ obeying

$$\forall\, \delta \in [-\delta_l, \delta_u]: \quad f(\mathbf{y} + \delta\mathbf{v}) = f(\mathbf{y}) \qquad \Rightarrow \qquad \mathbf{v}^{\mathrm{T}}\nabla f = 0 \tag{6}$$

and for the linear fitness landscape

$$\forall\, \kappa \in [-\kappa_l, \kappa_u]: \quad f(\mathbf{y} + \kappa\mathbf{v}) = f(\mathbf{y}) + \kappa f_{\mathbf{v}} = f(\mathbf{y}) + \kappa\mathbf{v}^{\mathrm{T}}\nabla f. \tag{7}$$

Note that Eq. (6) can be regarded as a special case of (7).

C. Behavior of SA-EAs on standard test fitness landscapes

Before we begin analysis of the real-parameter crossover operators, we would like to mention a number of test fitness landscapes where self-adaptive EAs have been largely applied.
1. A SA-EA should yield linear convergence order on sphere models, that is, one expects linearly falling curves of the logarithm of the residual distance of the population mean to the optimum (in the search space) with the number of generations, or mathematically

$$R(g) = R(0)\exp(-ag), \tag{8}$$

where a is the rate of decrease of $\log R$. This is what has been observed in self-adaptive ESs (For example, see [14]) and it seems reasonable to demand a similar behavior for other SA-EAs in unbounded search spaces. The efficiency of a SA-EA depends on how large the exponent a is. On sphere fitness function, there exist theoretical estimates of the parameter a for different self-adaptive ESs [15].
2. A SA-EA is expected to yield a linear divergence order on ridge models, where the optimum lies at infinity along a ridge axis. It has been shown experimentally that if the ridge axis coincides with a coordinate axis, a linear order of the following type:

$$r(g) = r(0)\exp(ag), \tag{9}$$

where $r(g)$ is a measure of distance traveled along the ridge axis by the population from a fixed point on the ridge at generation g. is possible to achieve with SA-EAs [10], whereas if the ridge axis is any arbitrary non-coordinate direction, a constant order is observed:

$$r(g) = r(0) + ag, \tag{10}$$

It is to be mentioned here that the performance behavior (10) violates our Proposition 3 postulated for functions obeying Eq. (7). It is an open question whether there exist SA-EAs that use parental population information only (that is, no further history), but exhibit an exponential behavior such like Eq. (9).
3. Besides these two extreme cases of convergence and divergence properties that a SA-EA should exhibit, they should also be able to adapt to a number of different landscapes [14], [10]:

Elliptic landscapes: Unlike in the sphere model, the emphasis of each coordinate direction is unequal in the fitness function. In these functions, a SA-EA is also expected to exhibit a linear order convergence.
Time-varying fitness landscapes: The fitness function changes when an EA population has converged sufficiently near the current optimum. This also tests a SA-EA's ability to find the optimum even when the initialization is done away from the optimum.
Fitness landscapes with non-zero correlations among variables: This tests an SA-EA's ability to adapt to fitness landscapes which are not linearly separable as functions of decision variables. Because of the non-zero correlations among the variables, a linear order convergence towards the optimum may be harder, but is desirable from a SA-EA. To this class of functions, the *rotated* versions of the elliptic landscapes and of the ridge functions belong.
Multi-modal fitness landscapes: A SA-EA should be able to avoid local optima and converge to the global optimum. After an initial transient of finding the global optimum, a linear order convergence is expected with a SA-EA.

Although the above are a few fitness landscapes where SA-EAs have been popularly tested, there may exist other landscapes which would test different adapting abilities of a SA-EA. We recognize the need of a more thorough study on test fitness landscape development for SA-EA, but we do not belabor it here. Instead, we focus our attention to analyze variation operators of a number of EAs and suggest when will each exhibit adequate self-adaptive properties. Because of the relative ease of doing the calculations, we first analyze the operators for flat fitness landscapes and later discuss or analyze the operators for linear and uni-modal fitness landscapes.

III. Analysis of Crossover Operators for Real-coded GAs in flat fitness landscapes

In this section, we analyze a number of crossover operators used in real-coded GAs. Specifically, the mean and variance of the children population are derived from the known distribution parameters of the parent population. These calculations will build up the basis for the determination of the dynamic behavior of the GAs considered. The theoretical predictions will be compared with simulations thereafter.

A. *Recombination operators of a special type*

Recombination in GAs is mainly inspired by the crossing over observed in nature. The standard procedure of recombination takes two parents and generates *two* offspring[3]. Since offspring are produced in pairs, we have $\mu/2$ pairs (that is, μ even is assumed) those individuals will be numbered by 1 and 2 and by an index counter k running from 1 to $\mu/2$.

The recombination operator proposed by Deb and Agrawal [8], called simulated binary crossover (SBX), is defined as:

$$\left\{ \begin{array}{l} y_{1,k} := \frac{1}{2}\left[(1-\beta_k)x_{1,k} + (1+\beta_k)x_{2,k}\right] \\ y_{2,k} := \frac{1}{2}\left[(1+\beta_k)x_{1,k} + (1-\beta_k)x_{2,k}\right]. \end{array} \right. \tag{11}$$

Here, $x_{1,k}$ and $x_{2,k}$ are independent samples from the parental population and β_k is a sample from a random number generator having the density

$$p(\beta) := \left\{ \begin{array}{ll} \frac{1}{2}(\eta+1)\beta^{\eta}, & \text{if} \quad 0 \le \beta \le 1 \\ \frac{1}{2}(\eta+1)\frac{1}{\beta^{\eta+2}}, & \text{if} \quad \beta > 1. \end{array} \right. \tag{12}$$

This distribution can easily be obtained from a uniform $\mathrm{u}(0,1)$ random number source by the transformation

$$\text{SBX}: \quad \beta(\mathrm{u}) := \left\{ \begin{array}{ll} (2\mathrm{u})^{\frac{1}{\eta+1}}, & \text{if} \quad \mathrm{u}(0,1) \le \frac{1}{2} \\ [2(1-\mathrm{u})]^{-\frac{1}{\eta+1}}, & \text{if} \quad \mathrm{u}(0,1) > \frac{1}{2}. \end{array} \right. \tag{13}$$

Note, the analysis to be performed is *not* restricted to the distribution (12). In addition we will investigate the BLX operator of Eshelman and Schaffer [7] and the so-called "fuzzy recombination" by Voigt, Mühlenbein, and Cvetković [6].

First, BLX is considered. By using the transformation

$$\beta = 2\xi - 1 \tag{14}$$

[3] From the algorithmic point of view, one might also consider operators producing only one child, as usual in Evolutions Strategies (see Section V).

applied to Eq. (11), one obtains

$$\left\{\begin{array}{llll} y_{1,k} := & (1-\xi_k)\, x_{1,k} & + & \xi_k\, x_{2,k} \\ y_{2,k} := & \xi_k\, x_{1,k} & + & (1-\xi_k)\, x_{2,k} \end{array}\right. \tag{15}$$

Now, choosing ξ uniformly from $[-\alpha, 1+\alpha]$, we arrive at Eshelman's and Schaffer's [7] blend crossover BLX-α

$$\text{BLX-}\alpha: \quad \xi := \mathrm{u}[-\alpha, 1+\alpha]. \tag{16}$$

The FR operator (fuzzy recombination) proposed by Voigt et al. [6] is similar to the SBX defined by (12) in the sense that it puts emphasize on the parental states: The probability density of the offspring y_i is maximal at the parental x_i. That is, the density of β in (11) has its maximum at $\beta = 1$. The only difference between SBX and FR is in the shape of $p(\beta)$. The shape of FR is just a triangle giving rise to call this recombination type "fuzzy". Using random numbers ζ obeying the triangular distribution density $p_\triangle(\zeta)$

$$p_\triangle(\zeta) := \left\{\begin{array}{lll} \zeta+1, & \text{if} & -1 \le \zeta < 0 \\ 1-\zeta, & \text{if} & 0 \le \zeta \le 1, \end{array}\right. \tag{17}$$

the β value to be used in (11) is obtained by the transformation

$$\text{“Fuzzy” recombination (FR)}: \quad \beta = 1 + 2\zeta d. \tag{18}$$

Here, d is a strategy parameter similar to α in (16). It determines how far offspring can be located from the parents. The random number ζ with triangle distribution (17) can be obtained by the standard method of inversion, or more easily, as the sum of two independent uniformly distributed random numbers $\mathrm{u}(0,1)$

$$\zeta := \mathrm{u}_1 + \mathrm{u}_2 - 1. \tag{19}$$

It is important to mention here that the above crossover operators can be considered as a hybrid of a $(\mu/2, \lambda)$-ES type 'intermediate' crossover operator followed by a perturbation operator – very similar to a mutation operator – whose variance depends on the difference in the parent solutions. This is obvious if we rewrite the first equation of (11) as follows:

$$y_{1,k} := (x_{1,k} + x_{2,k})/2 + \beta_k (x_{2,k} - x_{1,k})/2. \tag{20}$$

The distribution of the perturbation operator takes the shape of the distribution of β. For SBX and FR, this distribution has a non-zero mean and a mode (maximum) at $\beta = 1$, whereas for BLX, this distribution has a zero mean but is uniform within a certain interval. We argue that a variation operator which either uses directly the difference in parent solutions or uses an explicit evolution of a strategy parameter controlling the spread of children solutions (as in self-adaptive ESs) is essential for the resulting EA to exhibit self-adaptation. For optimum performance, it is then a matter of choosing the appropriate characteristic parameter of the underlying variation operator.

B. Definition of the variance measure

The crossover operator picks two individuals (considering only one component from the vectors) from the parental generation (g) and produces two offspring for generation $(g+1)$. For notational simplicity, let us denote an individual component by $x := x^{(g)}$ and the offspring by $y := x^{(g+1)}$. The population variance of the offspring is defined as [16]

$$\mathrm{Var}\{y\} := \frac{1}{\mu} \sum_{m=1}^{\mu} (y_m - \langle y \rangle)^2. \tag{21}$$

Here the subscript m is the individual counter running from one to the population size μ and $\langle y \rangle$ is the average (center of mass) individual. It is the aim of this section to calculate the *expected* value of $\mathrm{Var}\{y\}$, that is,

$$\sigma^2[y] := \mathrm{E}[\mathrm{Var}\{y\}] = \overline{\mathrm{Var}\{y\}}, \qquad \text{NB:} \quad \mathrm{E}[z] \equiv \overline{z}, \tag{22}$$

depending on the *expected* parental population variance, denoted by $\sigma^2[x]$ which is by definition

$$\sigma^2[x] := \mathrm{E}[(x - \overline{x})^2] = \overline{x^2} - \overline{x}^2. \tag{23}$$

C. Calculation of the population average $\langle y \rangle$

The population average is given by

$$\langle y \rangle := \frac{1}{\mu} \sum_{m=1}^{\mu} y_m . \tag{24}$$

Since the crossover operator produces two offspring by (11) one can re-arrange (24) to

$$\langle y \rangle := \frac{1}{\mu} \sum_{k=1}^{\mu/2} (y_{1,k} + y_{2,k}). \tag{25}$$

If one inserts (11) into (25) one easily obtains

$$\langle y \rangle := \frac{1}{\mu} \sum_{k=1}^{\mu/2} (x_{1,k} + x_{2,k}) = \frac{1}{\mu} \sum_{m=1}^{\mu} x_m . \tag{26}$$

Here it was taken into account that the $x_{1,k}$, $x_{2,k}$ are independent samples from the parental population and the x_m is just a renumbering of the individuals. Thus, the crossover operator preserves the population mean of the parental population. As discussed earlier, this behavior is considered as a desired property of crossover operators. Note, according to (11), for crossover operators (that is, producing *two* individuals) this also holds for each offspring pair produced independent of the distribution of the random variate β used. However, this does not necessarily hold for the expectation of a single offspring. In the latter case, it depends on the $p(\beta)$ density.

D. Calculation of the population variance

Starting from definition (21) one can alternatively write

$$\mathrm{Var}\{y\} := \frac{1}{\mu} \sum_{m=1}^{\mu} (y_m - \langle y \rangle)^2 = \frac{1}{\mu} \sum_{k=1}^{\mu/2} \left[(y_{1,k} - \langle y \rangle)^2 + (y_{2,k} - \langle y \rangle)^2 \right] . \tag{27}$$

As one can see, taking (11) and (26) into account, the $\mathrm{Var}\{y\}$ consists of a sum over $x_{i,k}x_{j,l}$ products. Since the final goal is to calculate $\sigma^2[y]$, the expected value is to be determined. This transfers to the calculation of the expected values of $x_{i,k}x_{j,l}$. Since the sampling process copies the individuals at random from the parental pool to the crossover operator, all $x_{i,k}$ are independent and identically distributed. Therefore one has

$$\overline{x_{i,k}x_{j,l}} := \begin{cases} \overline{x^2}, & \text{if} \quad i = k = j = l \\ \overline{x}^2, & \text{otherwise} \end{cases} \tag{28}$$

where $\overline{x}$ and $\overline{x^2}$ are the first and second order moments, respectively, of the parental population distribution. It is now the aim to re-arrange (27) in such a way that products $x_{i,k}x_{j,l}$ which $i = k = j = l$ are separated from the rest. To this end we consider the kth sum terms in (27). Using Eq. (11) and (26) one gets

$$\begin{aligned} y_{1,k} - \langle y \rangle &= \left(\frac{1}{2} - \frac{\beta_k}{2} \right) x_{1,k} + \left(\frac{1}{2} + \frac{\beta_k}{2} \right) x_{2,k} - \frac{1}{\mu} \sum_{l=1}^{\mu/2} (x_{1,l} + x_{2,l}) \\ &= \left(\frac{1}{2} - \frac{1}{\mu} - \frac{\beta_k}{2} \right) x_{1,k} + \left(\frac{1}{2} - \frac{1}{\mu} + \frac{\beta_k}{2} \right) x_{2,k} - \frac{1}{\mu} \sum_{l \neq k}^{\mu/2} (x_{1,l} + x_{2,l}) \end{aligned} \tag{29}$$

and

$$y_{2,k} - \langle y \rangle = \left(\frac{1}{2} - \frac{1}{\mu} + \frac{\beta_k}{2} \right) x_{1,k} + \left(\frac{1}{2} - \frac{1}{\mu} - \frac{\beta_k}{2} \right) x_{2,k} - \frac{1}{\mu} \sum_{l \neq k}^{\mu/2} (x_{1,l} + x_{2,l}). \tag{30}$$

Taking the square yields

$$\begin{aligned}(y_{1,k}-\langle y\rangle)^2 \;&=\; \left(\frac{1}{2}-\frac{1}{\mu}-\frac{\beta_k}{2}\right)^2 x_{1,k}^2+\left(\frac{1}{2}-\frac{1}{\mu}+\frac{\beta_k}{2}\right)^2 x_{2,k}^2+\frac{1}{\mu^2}\left(\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l})\right)^2\\ &+2\left[\left(\frac{1}{2}-\frac{1}{\mu}\right)^2-\frac{\beta_k^2}{4}\right]x_{1,k}x_{2,k}-2\left(\frac{1}{2}-\frac{1}{\mu}-\frac{\beta_k}{2}\right)x_{1,k}\frac{1}{\mu}\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l})\\ &-2\left(\frac{1}{2}-\frac{1}{\mu}+\frac{\beta_k}{2}\right)x_{2,k}\frac{1}{\mu}\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l}) \qquad (31)\end{aligned}$$

and

$$\begin{aligned}(y_{2,k}-\langle y\rangle)^2 \;&=\; \left(\frac{1}{2}-\frac{1}{\mu}+\frac{\beta_k}{2}\right)^2 x_{1,k}^2+\left(\frac{1}{2}-\frac{1}{\mu}-\frac{\beta_k}{2}\right)^2 x_{2,k}^2+\frac{1}{\mu^2}\left(\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l})\right)^2\\ &+2\left[\left(\frac{1}{2}-\frac{1}{\mu}\right)^2-\frac{\beta_k^2}{4}\right]x_{1,k}x_{2,k}-2\left(\frac{1}{2}-\frac{1}{\mu}+\frac{\beta_k}{2}\right)x_{1,k}\frac{1}{\mu}\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l})\\ &-2\left(\frac{1}{2}-\frac{1}{\mu}-\frac{\beta_k}{2}\right)x_{2,k}\frac{1}{\mu}\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l}). \qquad (32)\end{aligned}$$

Putting (31) and (32) together, one obtains

$$\begin{aligned}&(y_{1,k}-\langle y\rangle)^2+(y_{2,k}-\langle y\rangle)^2\\ &=\; 2\left[\left(\frac{1}{2}-\frac{1}{\mu}\right)^2+\frac{\beta_k^2}{4}\right](x_{1,k}^2+x_{2,k}^2)+\frac{2}{\mu^2}\left(\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l})\right)^2\\ &\quad+4\left[\left(\frac{1}{2}-\frac{1}{\mu}\right)^2-\frac{\beta_k^2}{4}\right]x_{1,k}x_{2,k}-4\left(\frac{1}{2}-\frac{1}{\mu}\right)x_{1,k}\frac{1}{\mu}\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l})\\ &\quad-4\left(\frac{1}{2}-\frac{1}{\mu}\right)x_{2,k}\frac{1}{\mu}\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l}) \qquad (33)\end{aligned}$$

In order to proceed, we calculate the $(\cdots)^2$ term in the second line of (33)

$$\begin{aligned}&\left(\sum_{l\neq k}^{\mu/2}(x_{1,l}+x_{2,l})\right)^2\\ &=\; \sum_{l\neq k}^{\mu/2}\sum_{m\neq k}^{\mu/2}(x_{1,l}+x_{2,l})(x_{1,m}+x_{2,m})\\ &=\; \sum_{l\neq k}^{\mu/2}\sum_{m\neq k}^{\mu/2}x_{1,l}x_{1,m}+x_{1,l}x_{2,m}+x_{2,l}x_{1,m}+x_{2,l}x_{2,m}\\ &=\; \sum_{l\neq k}^{\mu/2}\sum_{m\neq k}^{\mu/2}(x_{1,l}x_{1,m}+x_{2,l}x_{2,m})+\sum_{l\neq k}^{\mu/2}\sum_{m\neq k}^{\mu/2}x_{1,l}x_{2,m}+\sum_{l\neq k}^{\mu/2}\sum_{m\neq k}^{\mu/2}x_{1,m}x_{2,l}\\ &=\; \sum_{l\neq k}^{\mu/2}\sum_{m\neq k}^{\mu/2}(x_{1,l}x_{1,m}+x_{2,l}x_{2,m})+2\sum_{l\neq k}^{\mu/2}\sum_{m\neq k}^{\mu/2}x_{1,l}x_{2,m}\\ &=\; \sum_{l\neq k}^{\mu/2}(x_{1,l}^2+x_{2,l}^2)+\sum_{l\neq k}^{\mu/2}\sum_{m\neq l}^{\mu/2}(x_{1,l}x_{1,m}+x_{2,l}x_{2,m})+2\sum_{l\neq k}^{\mu/2}\sum_{m\neq k}^{\mu/2}x_{1,l}x_{2,m}. \qquad (34)\end{aligned}$$

Inserting this into (33) gives

$$
\begin{aligned}
&(y_{1,k} - \langle y \rangle)^2 + (y_{2,k} - \langle y \rangle)^2 \\
&= 2\left[\left(\frac{1}{2} - \frac{1}{\mu}\right)^2 + \frac{\beta_k^2}{4}\right](x_{1,k}^2 + x_{2,k}^2) + \frac{2}{\mu^2}\sum_{l \neq k}^{\mu/2}(x_{1,l}^2 + x_{2,l}^2) \\
&\quad +4\left[\left(\frac{1}{2} - \frac{1}{\mu}\right)^2 - \frac{\beta_k^2}{4}\right] x_{1,k}x_{2,k} - 4\left(\frac{1}{2} - \frac{1}{\mu}\right)\frac{1}{\mu}\sum_{l \neq k}^{\mu/2}(x_{1,k}x_{1,l} + x_{1,k}x_{2,l} + x_{2,k}x_{1,l} + x_{2,k}x_{2,l}) \\
&\quad +\frac{2}{\mu^2}\sum_{l \neq k}^{\mu/2}\sum_{m \neq l}^{\mu/2}(x_{1,l}x_{1,m} + x_{2,l}x_{2,m}) + \frac{4}{\mu^2}\sum_{l \neq k}^{\mu/2}\sum_{m \neq k}^{\mu/2} x_{1,l}x_{2,m}.
\end{aligned} \tag{35}
$$

In order to calculate (22), the expectation of the bracket in (27) must be determined. Using (35) and (28) one obtains

$$
\begin{aligned}
&\mathrm{E}\left[(y_{1,k} - \langle y \rangle)^2 + (y_{2,k} - \langle y \rangle)^2\right] \\
&= \left[4\left(\frac{1}{2} - \frac{1}{\mu}\right)^2 + \overline{\beta_k^2}\right]\overline{x^2} + \frac{4}{\mu^2}\overline{x^2}\sum_{l \neq k}^{\mu/2} 1 \\
&\quad +\left[4\left(\frac{1}{2} - \frac{1}{\mu}\right)^2 - \overline{\beta_k^2}\right]\overline{x}^2 - 4\left(\frac{1}{2} - \frac{1}{\mu}\right)\frac{1}{\mu}\overline{x}^2\sum_{l \neq k}^{\mu/2}(1+1+1+1) \\
&\quad +\frac{2}{\mu^2}\overline{x}^2\sum_{l \neq k}^{\mu/2}\sum_{m \neq l}^{\mu/2}(1+1) + \frac{4}{\mu^2}\overline{x}^2\sum_{l \neq k}^{\mu/2}\sum_{m \neq k}^{\mu/2} 1.
\end{aligned} \tag{36}
$$

Collecting all $\overline{x^2}$ and $\overline{x}^2$ terms yields a remarkably simple result

$$
\begin{aligned}
&\mathrm{E}\left[(y_{1,k} - \langle y \rangle)^2 + (y_{2,k} - \langle y \rangle)^2\right] \\
&= \left[\overline{\beta_k^2} + 4\left(\frac{1}{2} - \frac{1}{\mu}\right)^2 + \left(\frac{\mu}{2} - 1\right)\frac{4}{\mu^2}\right]\overline{x^2} \\
&\quad -\left[\overline{\beta_k^2} - 4\left(\frac{1}{2} - \frac{1}{\mu}\right)^2 + 16\left(\frac{1}{2} - \frac{1}{\mu}\right)\frac{1}{\mu}\left(\frac{\mu}{2} - 1\right) - \frac{4}{\mu^2}\left(\frac{\mu}{2} - 1\right)\left(\frac{\mu}{2} - 2\right) - \frac{4}{\mu^2}\left(\frac{\mu}{2} - 1\right)^2\right]\overline{x}^2 \\
&= \left[1 - \frac{2}{\mu} + \overline{\beta_k^2}\right]\left(\overline{x^2} - \overline{x}^2\right).
\end{aligned} \tag{37}
$$

After back-substitution in (22) taking (27) into account one obtains

$$
\begin{aligned}
\sigma^2[y] &= \frac{1}{\mu}\sum_{k=1}^{\mu/2}\mathrm{E}\left[(y_{1,k} - \langle y \rangle)^2 + (y_{2,k} - \langle y \rangle)^2\right] \\
&= \frac{1}{\mu}\sum_{k=1}^{\mu/2}\left[1 - \frac{2}{\mu} + \overline{\beta_k^2}\right]\left(\overline{x^2} - \overline{x}^2\right) \\
&= \frac{1}{\mu}\frac{\mu}{2}\left[1 - \frac{2}{\mu} + \overline{\beta_k^2}\right]\left(\overline{x^2} - \overline{x}^2\right).
\end{aligned} \tag{38}
$$

Since the β_k are samples from the same distribution, we write $\overline{\beta_k^2} = \overline{\beta^2}$; furthermore (23) is taken into account. Thus, we finally arrive at the *evolution equation of the expected population variance*

$$
\sigma^{2\,(g+1)} = \frac{1}{2}\left(1 - \frac{2}{\mu} + \overline{\beta^2}\right)\sigma^{2\,(g)}. \tag{39}
$$

Its solution by recurrence leads to the equation of expected variance dynamics

$$
\sigma^2[y]^{(g)} = \sigma^2[y]^{(0)}\left(\frac{1}{2} - \frac{1}{\mu} + \frac{\overline{\beta^2}}{2}\right)^g, \tag{40}
$$

that is, one observes an exponential change of the population variance. Collapsing or exploding of the population is thus determined by μ and $\overline{\beta^2}$. The latter will be determined for some typical crossover operators below.

E. Typical Crossover Operators

The evolution equation (38) contains the $\overline{\beta^2}$ which depends on the random number distribution used. By definition, we have

$$\overline{\beta^2} = \int \beta^2 p(\beta) \mathrm{d}\beta. \tag{41}$$

For SBX and BLX the standard way of generating β is via transformation from the uniform variate $u = \mathrm{u}(0,1)$ one can alternatively write

$$\overline{\beta^2} = \int_{u=0}^{u=1} (\beta(u))^2 \mathrm{d}u. \tag{42}$$

As to FR, Eq. (17) has to be used in order to calculate $\overline{\beta^2}$

$$\overline{\beta^2} = \int_{\zeta=-1}^{\zeta=1} (\beta(\zeta))^2 p_{\triangle}(\zeta) \mathrm{d}\zeta. \tag{43}$$

E.1 SBX-η

The SBX uses a random number distribution according to (13). The calculation of the second moment $\overline{\beta^2}$ is straightforward. Using (42) one has with (13)

$$\overline{\beta^2} = \int_{u=0}^{u=1/2} (2u)^{\frac{2}{\eta+1}} \mathrm{d}u + \int_{u=1/2}^{u=1} [2(1-u)]^{-\frac{2}{\eta+1}} \mathrm{d}u. \tag{44}$$

After substitution $s := 2u$ and $t := 2(1-u)$ for the first and the second integral, respectively, one ends up with the following equation for $\eta \neq 1$:

$$\overline{\beta^2} = \frac{1}{2} \left[\frac{1}{1+\frac{2}{\eta+1}} s^{\left(1+\frac{2}{\eta+1}\right)} \right]_0^1 + \frac{1}{2} \left[\frac{1}{1-\frac{2}{\eta+1}} t^{\left(1-\frac{2}{\eta+1}\right)} \right]_0^1 \tag{45}$$

which gives for $\eta > 1$

$$\overline{\beta^2} = \frac{1}{2} \left[\frac{\eta+1}{\eta+1+2} + \frac{\eta+1}{\eta+1-2} \right], \tag{46}$$

and finally

$$\overline{\beta^2} = \frac{(\eta+1)^2}{(\eta+3)(\eta-1)}. \tag{47}$$

The case $\eta \leq 1$ is not covered by the above equation. From Eq. (45), one immediately obtains $\overline{\beta^2} \to \infty$, that is, the second moment does not exist. In practice, however, the evolution of variance is restricted by the finite number representation of a computer.

Eq. (47) leads with (40) to the variance dynamics of the SBX operator for $\eta > 1$

$$\sigma^2[y]^{(g)} = \sigma^2[y]^{(0)} \left(1 + \frac{2}{(\eta+3)(\eta-1)} - \frac{1}{\mu} \right)^g. \tag{48}$$

As one can see, depending on the choice of η, the expected population can increase or contract for a given population size μ. Eq. (48) can also be used to determine the minimal population size given a fixed η. Since variance increase is the desired behavior (see Proposition 2 in Section II-B), the terms in the large parenthesis of (48) must be greater than 1. Therefore, one obtains by simple calculation

$$\text{population size:} \quad \mu > \frac{1}{2}(\eta+3)(\eta-1). \tag{49}$$

E.2 BLX-α

Starting from Eq. (41) taking (16), that is, $p(\xi) = 1/(1+2\alpha)$, and (14) into account, one obtains the integral

$$\begin{aligned}\overline{\beta^2} &= \frac{1}{1+2\alpha}\int_{\xi=-\alpha}^{\xi=1+\alpha}(2\xi-1)^2\mathrm{d}\xi \\ &= \frac{1}{1+2\alpha}\int_{\xi=-\alpha}^{\xi=1+\alpha}(4\xi^2-4\xi+1)\mathrm{d}\xi \\ &= \frac{1}{1+2\alpha}\left[\frac{4}{3}\xi^3-2\xi^2+\xi\right]_{-\alpha}^{1+\alpha}.\end{aligned} \tag{50}$$

After some simple calculations one gets the result

$$\overline{\beta^2} = \frac{1}{3}(2\alpha+1)^2. \tag{51}$$

Thus we get with (39) the evolution equation

$$\sigma^2[y]^{(g)} = \sigma^2[y]^{(0)}\left(1+\frac{1}{3}(2\alpha^2+2\alpha-1)-\frac{1}{\mu}\right)^g. \tag{52}$$

Again, the contraction/expansion of the population depends on the choice of α. E.g., $\alpha = 0$ (BLX-0.0) leads immediately to a contracting population, no matter how μ is chosen.

In [7] Eshelman and Schaffer suggested to use $\alpha = 1/2$ (BLX-0.5) because: "Only when $\alpha = 0.5$ does the probability that an offspring will lie outside its parents become equal to the probability that it will lie between its parents." Furthermore they inferred [7, p. 193]: "In other words, $\alpha = 0.5$ balances the convergent and divergent tendencies in the absence of selection pressure." Thought their former statement is correct, the latter one is wrong. As one can easily see from Eq. (52) for $\alpha = 1/2$, the variance ratio becomes

$$\text{BLX-0.5:} \quad \frac{\sigma^{2\,(g+1)}}{\sigma^{2\,(g)}} = \frac{7}{6} - \frac{1}{\mu}. \tag{53}$$

That is, depending on μ the population converges or diverges. A simple calculation yields:

$$\text{BLX-0.5 divergence:} \quad \mu > 6, \tag{54}$$

that is, only for $\mu = 6$ convergent and divergent tendencies are balanced by BLX-0.5. This result is valid for GAs with random sampling, where the parents chosen for crossover are drawn at random from the parental pool of size μ. The CHC-GA of Eshelman [17] uses a more sophisticated selection technique in order to prevent incest. Even in this case, there still remains a random sampling component producing similar results for the σ^2 dynamics. Furthermore, even a fictitious algorithm without diversity loss, as in the case of infinite population size μ, does not confirm the claim of [7]. However, the real variance evolution of CHC remains still to be analyzed.

Since it is postulated for self-adaptation to have a diverging population in flat fitness landscapes (see Proposition 2 in Section II-B), we may ask how to choose α given a fixed population size μ. Divergence is obviously fulfilled if the large parenthesis in (52) is greater than 1. Resolving for α one obtains

$$\alpha > \frac{\sqrt{3+6/\mu}-1}{2}. \tag{55}$$

Similar to (49) one can ask for the minimal population size. By the criterion $\sigma^2[y]^{(g)} > \sigma^2[y]^{(0)}$ applied to (52) one finds

$$\text{population size:} \quad \mu > \frac{3}{2\alpha+2\alpha^2-1}. \tag{56}$$

E.3 FR-d (fuzzy recombination)

In case of fuzzy recombination, the calculation of $\overline{\beta^2}$ starts from (43) using (17) and (18)

$$\overline{\beta^2} = \int_{\zeta=-1}^{\zeta=0} (1+2\zeta d)^2(\zeta+1)\mathrm{d}\zeta + \int_{\zeta=0}^{\zeta=1} (1+2\zeta d)^2(1-\zeta)\mathrm{d}\zeta. \tag{57}$$

The integration can be carried out easily. One finally obtains

$$\overline{\beta^2} = 1 + \frac{2}{3}d^2. \tag{58}$$

This yields with (40) the variance dynamics of the FR operator

$$\sigma^2[y]^{(g)} = \sigma^2[y]^{(0)} \left(1 + \frac{1}{3}d^2 - \frac{1}{\mu}\right)^g. \tag{59}$$

We can now determine the minimum population size by demanding $\sigma^2[y]^{(g)} > \sigma^2[y]^{(0)}$ for flat fitness landscapes. One immediately gets

$$\text{population size:} \quad \mu > \frac{3}{d^2}. \tag{60}$$

In [6, p. 106] Voigt et al. suggested $d \geq 1/2$ and as a rule of thumb to use $d = 1/2$. As one can see from (60) the case $d = 1/2$ requires $\mu > 12$. From the viewpoint of variance evolution in flat fitness landscapes, there is no reason to postulate $d \geq 1/2$. However, μ must scale quadratically with the decrease of d. Interestingly, Voigt et al. realized that there is a critical population size (which they have called N^* in [6, p. 107]) below which the GA does not work appropriately: "*It depends om* [should read 'on'] I, n *and* ϵ." However, they did not realize that this population size depends also on d.

F. Some simulation examples

Figure 1 compares the formula of the population variance dynamics with simulations on the SBX operator. Figure 2 shows results for the BLX operator. Figure 3 displays simulations for the "fuzzy" recombination operator. As one can see, each of these operators can exhibit exponentially increasing as

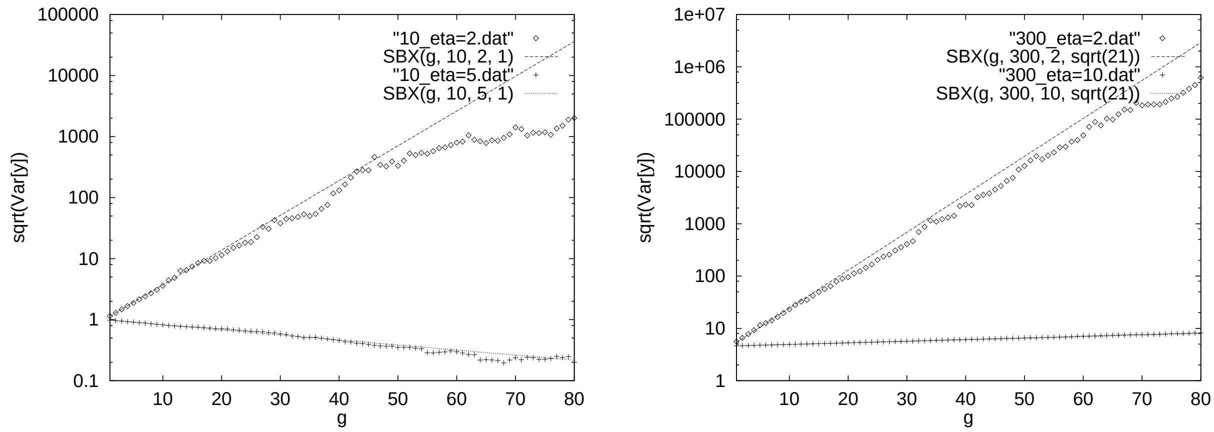

Fig. 1. Evolution of the square root of the expected population variance for the SBX operator. Left picture: population size 10 with $\eta = 2$ (upper curve) and $\eta = 5$ (lower curve). Right picture: population size 300 with $\eta = 2$ (upper curve) and $\eta = 10$ (lower curve). The simulation results have been obtained by averaging over 100 (pop-size = 300) and 3000 (pop-size = 10), respectively, independent runs.

well as decreasing population variance, depending on the choice of corresponding characteristic parameter value (η, α, or d, respectively). The deviation of the simulation from the theoretical predictions are due to cumulation of statistical errors.

IV. Simulation Results on Sphere Model

The above calculations on population variance on flat fitness allow us to investigate if GAs with different crossover operators but having similar variance property have similar performance on the sphere model.

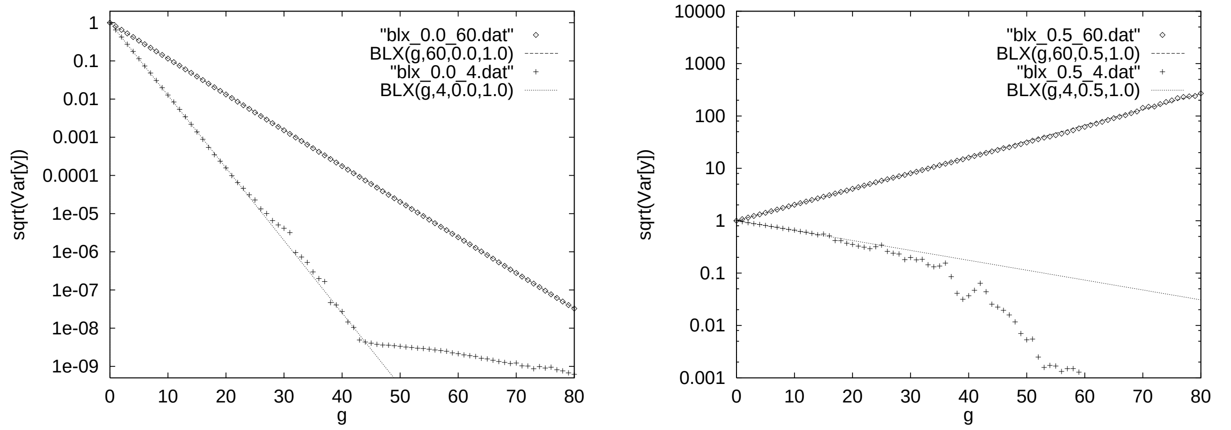

Fig. 2. Evolution of the square root of the expected population variance for the BLX operator. Left picture: BLX-0.0 for population size 60 (upper curve) and 4 (lower curve). Right picture: BLX-0.5 for population size 60 (upper curve) and 4 (lower curve). The simulation results have been obtained by averaging over 100 (pop-size = 60) and 2000 (pop-size = 4), respectively, independent runs.

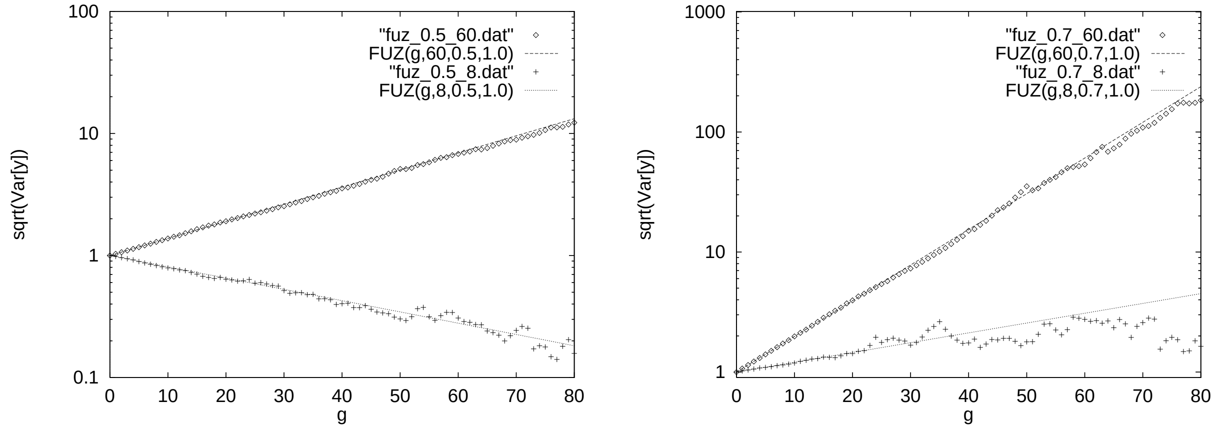

Fig. 3. Evolution of the square root of the expected population variance for the fuzzy recombination operator. Left picture: $d = 0.5$ for population size 60 (upper curve) and 8 (lower curve). Right picture: $d = 0.7$ for population size 60 (upper curve) and 8 (lower curve). The simulation results have been obtained by averaging over 100 (pop-size = 60) and 3000 (pop-size = 8), respectively, independent runs.

We realize that ideally population variance under a complete GA cycle (reproduction and variation) must be made on sphere model to compare the performances of different GAs. But, such an analysis is certainly diffcult, although may not be intractable. In any case, the above calculations provide some guidelines for GAs with different crossover implementations to be compared in any fitness landscape.

Equating the variance terms (Eqs. (48), (52), and (59)) for the flat fitness case, we obtain the parameter values for three crossover operators, given in Table I.

In the experimental results, we use a 20-dimensional sphere function. Along with the respective crossover operators, we use a binary tournament selection and no mutation. A population size of 100 is chosen. We have initialized the population at $x_i \in [-5, 5]$. For three crossover operators, we choose the parameter setting, presented in the first row of Table I, although they are calculated for flat fitness landscapes. Figure 4 shows the distance of the best solution in the population from the optimum for all three crossover operators (averaged over 10 independent runs). It is clear from the plot that GAs with all three crossover operators have more or less similar performance. This is also demonstrated in Figure 5, which plots the standard deviation in the arbitrarily chosen variable x_{10}.

TABLE I

PARAMETER VALUES FOR IDENTICAL POPULATION VARIANCE GROWTH IN THREE CROSSOVER OPERATORS.

SBX η	BLX α	FR d
2	0.662	1.095
3	0.500	0.707
5	0.419	0.433
10	0.381	0.226

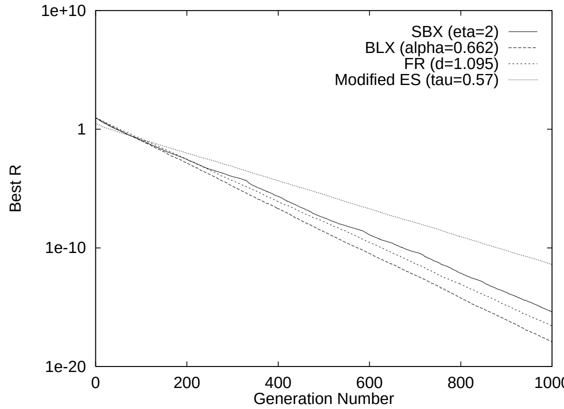

Fig. 4. Population best distance from the optimum. The curve labeled 'Modified ES' is described in Section VI.

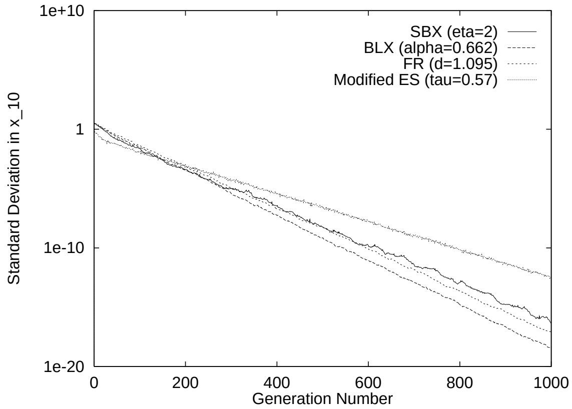

Fig. 5. Standard deviation in x_{10}. The curve labeled 'Modified ES' is described in Section VI.

V. ANALYSIS OF SELF-ADAPTIVE ES IN FLAT FITNESS LANDSCAPES

A. Mutation and recombination operators

The standard ES algorithm using isotropic Gaussian mutations has the following update rule

$$\forall\, l = 1,\ldots,\lambda: \quad \left\{ \begin{array}{l} \sigma_l^{(g+1)} := \mathrm{Reco}_\sigma(\sigma_{1;\lambda}^{(g)},\ldots,\sigma_{\mu;\lambda}^{(g)}) \cdot \Xi_\sigma \\ \mathbf{y}_l^{(g+1)} := \mathrm{Reco}_{\mathbf{y}}(\mathbf{y}_{1;\lambda}^{(g)},\ldots,\mathbf{y}_{\mu;\lambda}^{(g)}) + \sigma_l^{(g+1)}\vec{\mathcal{N}}(0,1). \end{array} \right. \tag{61}$$

The parents are selected by truncation, that is, taking the μ best (as to their fitness) out of the λ offspring from the previous generation, denoted by $\sigma_{m;\lambda}^{(g)}$ and $\mathbf{y}_{m;\lambda}^{(g)}$, respectively.

Since we are interested in the flat fitness landscape behavior, the evolution of the strategy parameter $\sigma^{(g)}$ is decoupled from the evolution of the object parameters. That is, the population variance of a single object parameter entry $y_i := (\mathbf{y})_i$ after selection is given by (writing $y = y_i$)

$$\sigma^2[y]^{(g+1)} = \mathrm{E}\left[\mathrm{Var}\left\{\mathrm{Reco}_{\mathbf{y}}(\mathbf{y}_{1;\lambda}^{(g)},\ldots,\mathbf{y}_{\mu;\lambda}^{(g)})\right\}\right] + \mathrm{E}\left[\sigma^{2\;(g+1)}\right]. \tag{62}$$

This follows immediately from the second line in (61). The population variance is the sum of the (independent) population variances after recombination and mutation.

The recombinational variance depends on the way how recombination is performed. Usually, in ESs ρ parental individuals contribute to the procreation of *one* offspring individual (called recombinant, for an introduction, see e.g. [18, p. 83]). This is in contrast to standard crossover in GAs. In this paper we will restrict our analysis to $\rho = \mu$ multi-recombination often used in applications. We will consider intermediate and dominant recombination as well.

If there is (μ/μ_I)-intermediate recombination, the center of mass of all μ parents is calculated. Therefore, there is no variance at all, since all recombinants are the same. However, if dominant (μ/μ_D) recombination is considered, a random sampling is performed: Given the expected parental population variance $\sigma^2[y]^{(g)}$, the variance of the recombinants becomes $(1-1/\mu)\sigma^2[y]^{(g)}$ [16]. Thus, we have

$$\mathrm{E}\left[\mathrm{Var}\left\{\mathrm{Reco}_{\mathbf{y}}(\mathbf{y}_{1;\lambda}^{(g)},\ldots,\mathbf{y}_{\mu;\lambda}^{(g)})\right\}\right] = \left\{ \begin{array}{rl} 0, & \text{if} \quad (\mu/\mu_I,\lambda)\text{-ES} \\ \left(1-\frac{1}{\mu}\right)\sigma^2[y]^{(g)}, & \text{if} \quad (\mu/\mu_D,\lambda)\text{-ES.} \end{array} \right. \tag{63}$$

Other recombination variants might be considered, however, it is quite clear that for standard ES recombination techniques

$$0 \le \mathrm{E}\left[\mathrm{Var}\left\{\mathrm{Reco}_{\mathbf{y}}(\mathbf{y}^{(g)}_{1;\lambda}, \ldots, \mathbf{y}^{(g)}_{\mu;\lambda})\right\}\right] \le \sigma^2[y]^{(g)} \tag{64}$$

holds, that is, the variance can only reduce or remains unchanged by the recombination operator. The main contribution to the population variance, however, is given by the mutations, those standard deviation is computed in the first line of (61).

There are different ways to perform the recombination and mutation on the strategy parameters. The usual way of recombination is the intermediate (μ/μ_I) one, resulting in an average recombinant individual. This is accomplished by calculating the arithmetic mean of the μ parental σ values

$$\text{intermediate } (\mu/\mu_I)\text{-recombination:} \quad \mathrm{Reco}_{\sigma_I} := \frac{1}{\mu}\sum_{m=1}^{\mu} \sigma^{(g)}_{m;\lambda} =: \sigma^{(g+1)}_{r_I}. \tag{65}$$

Alternatively one can use the geometric mean

$$\text{geometric } (\mu/\mu_G)\text{-recombination:} \quad \mathrm{Reco}_{\sigma_G} := \sqrt[\mu]{\prod_{m=1}^{\mu} \sigma^{(g)}_{m;\lambda}} =: \sigma^{(g+1)}_{r_G}. \tag{66}$$

Usually the former (65) is the standard way of recombining the mutation strength. Interestingly, the dominant (or discrete) recombination often used for object parameters has proved to be not suited for strategy parameters. Therefore, it will be not considered here. Alternatively, the geometric version (66) has been suggested by Hansen [19]. He claims that geometric recombination is a "drift-free" operator in contrast to intermediate (arithmetic) (μ/μ_I)-recombination. The latter should exhibit a systematic drift toward larger σ values. As we will see below, this statement of Hansen [19, p. 29] is not correct.

The change of the mutation strength σ is done by multiplicative mutations denoted by Ξ_σ in the first line of (61). This operator can be realized in different ways, e.g.

$$\Xi_\sigma := \exp(\zeta), \tag{67}$$

with a random variate ζ. The most prominent version uses $\mathcal{N}(0, \tau^2)$ normally distributed random numbers ζ [20]

$$\zeta_{LN} := \tau\mathcal{N}(0,1). \tag{68}$$

That is, Ξ is log-normally distributed and the mutation operator (67) is called the log-normal operator. Besides other mutation rules discussed in [15], the symmetric two-point rule is often used [21, p. 47]. The symmetric version reads

$$\zeta_{TP} := \begin{cases} +\varepsilon, & \text{if} \quad \mathrm{u}[0,1] \le \frac{1}{2} \\ -\varepsilon, & \text{if} \quad \mathrm{u}[0,1] > \frac{1}{2} \end{cases}, \quad \varepsilon > 0. \tag{69}$$

Therefore, ζ_{TP} yields e^{ε} and $\mathrm{e}^{-\varepsilon}$ with the same probability 1/2. Note, usually this rule is rewritten by the substitution $\alpha := \mathrm{e}^{\varepsilon}$ in terms of

$$\Xi_{TP} := \begin{cases} \alpha, & \text{if} \quad \mathrm{u}[0,1] \le \frac{1}{2} \\ 1/\alpha, & \text{if} \quad \mathrm{u}[0,1] > \frac{1}{2}. \end{cases} \tag{70}$$

B. Calculation of the population variance

The recursive equation of the population variance can be derived from (62). Since the first term in (62) is given by (63), the calculation of $\mathrm{E}[\sigma^{2\,(g+1)}]$ remains to be done. Here, it is important to realize that due to the selection neutrality in flat fitness landscapes, the first line in (61) is decoupled from the second line. That is, the σ evolution is fully determined by the first line in (61). Therefore it suffices to calculate the expected variance produced by the recombination operator in the first line of (61). This will be done in the next two points; furthermore we will see that the evolution of the expected σ_r (r stands for the recombinant) and σ_r^2 is different, that is, $\overline{\sigma_r^{2\,(g)}} \ne \overline{\sigma_r^{(g)}}^2$. After that, the calculation of the expected population variance $\sigma^2[y]^{(g)}$ will be performed.

B.1 Determining $\overline{\sigma_r}$

In order to calculate the expected value of the strategy parameter σ after recombination in the first line of (61), denoted by $\overline{\sigma_r}$, we have to take (67) into account. The $\sigma_{m;\lambda}^{(g)}$ are generated according to the first line in (61), (65), and (66), respectively, that is, the σ value of the lth offspring is obtained as

$$\sigma_l^{(g)} := \sigma_r^{(g)} \mathrm{e}^{\zeta_l^{(g)}}. \tag{71}$$

Since selection does not prefer any instantiation of (71), the recombination result using μ individuals becomes

$$\text{intermediate } (\mu/\mu_I)\text{-recombination:} \quad \sigma_{r_I}^{(g+1)} := \frac{1}{\mu} \sum_{m=1}^{\mu} \sigma_{r_I}^{(g)} \mathrm{e}^{\zeta_m^{(g)}} \tag{72}$$

and

$$\text{geometric } (\mu/\mu_G)\text{-recombination:} \quad \sigma_{r_G}^{(g+1)} := \sqrt[\mu]{\prod_{m=1}^{\mu} \sigma_{r_G}^{(g)} \mathrm{e}^{\zeta_m^{(g)}}}. \tag{73}$$

Determining the expected value taking the statistical independence into account, one gets for the (μ/μ_I) version

$$\overline{\sigma_{r_I}^{(g+1)}} = \overline{\sigma_{r_I}^{(g)}} \frac{1}{\mu} \sum_{m=1}^{\mu} \overline{\mathrm{e}^{\zeta_m^{(g)}}} = \overline{\sigma_{r_I}^{(g)}}\ \overline{\mathrm{e}^{\zeta}} \tag{74}$$

and for the geometric recombination

$$\overline{\sigma_{r_G}^{(g+1)}} = \overline{\sigma_{r_G}^{(g)}}\ \mathrm{E}\left[\prod_{m=1}^{\mu} \exp\left(\frac{\zeta_m^{(g)}}{\mu}\right)\right] = \overline{\sigma_{r_G}^{(g)}} \left(\overline{\mathrm{e}^{\zeta/\mu}}\right)^{\mu}. \tag{75}$$

The expectations $\overline{\mathrm{e}^{\zeta}}$ and $\overline{\mathrm{e}^{\zeta/\mu}}$ can be easily calculated for the log-normal and the two-point case (for a summary, see Table III). Since the expectation of the log-normal variate $\overline{\exp(\gamma\mathcal{N}(0,1))} = \exp(\gamma^2/2)$, we have

$$\text{log-normal mutation:} \quad \overline{\mathrm{e}^{\zeta_{LN}}} = \mathrm{e}^{\tau^2/2} \quad \text{and} \quad \overline{\mathrm{e}^{\zeta_{LN}/\mu}} = \mathrm{e}^{\tau^2/2\mu^2}. \tag{76}$$

For the symmetric two-point rule (69) we immediately find

$$\text{two-point mutation:} \quad \overline{\mathrm{e}^{\zeta_{TP}}} = \frac{1}{2}(\mathrm{e}^{\varepsilon} + \mathrm{e}^{-\varepsilon}) = \cosh(\varepsilon) \quad \text{and} \quad \overline{\mathrm{e}^{\zeta_{TP}/\mu}} = \cosh(\varepsilon/\mu). \tag{77}$$

From Eqn. (74/75) one can easily derive the evolution equations, that is, the g-dynamics of $\overline{\sigma_r}$. By iteration one immediately obtains the results displayed in Table II. As one can see, as long as $|\tau| > 0$ and $|\varepsilon| > 0$,

TABLE II

Dynamics of the expected value of σ after recombination. $\overline{\sigma_r^{(0)}}$ is the initial value of mutation strength.

	(μ/μ_I)-recombination	(μ/μ_G)-recombination
log-normal rule	$\overline{\sigma_r^{(g)}} = \overline{\sigma_r^{(0)}} \exp\left(\frac{\tau^2}{2} g\right)$	$\overline{\sigma_r^{(g)}} = \overline{\sigma_r^{(0)}} \exp\left(\frac{\tau^2}{2\mu} g\right)$
two-point rule	$\overline{\sigma_r^{(g)}} = \overline{\sigma_r^{(0)}} [\cosh(\varepsilon)]^g$	$\overline{\sigma_r^{(g)}} = \overline{\sigma_r^{(0)}} \left[\cosh\left(\frac{\varepsilon}{\mu}\right)\right]^{\mu g}$

respectively, the expected value of σ increases exponentially with g. This is a bit astonishing for geometric recombination, because according to Hansen [19, p. 29], geometric recombination should be "drift-free" in contrast to the standard intermediate one. As we see here, the type of recombination does not introduce any drift. The drift comes from the mutation operators. Both the log-normal and the symmetric two-point rule are biased toward a σ increase, because both the expected values (76) and (77) are greater than one. Looking at (74) and (75), however, we see that the averaging crossover conserves the expected effect of the mutation operator used, whereas the geometric version reduces the effect of the mutations.This effect is also observed in the variance evolution $\sigma^2[y]^{(g)}$ of the parental object parameters (see Section V-B.3 below and Figure 6).

As we have already pointed out, a slight increase of the expected σ is desired for a correct working in flat fitness landscapes. However, one can also construct σ mutation operators with $\overline{e^\zeta} = 1$. This would guarantee that $\overline{\sigma_r^{(g)}}$ remains constant. Interestingly, even for that case, the expected population variance $\sigma^2[y]^{(g)}$ can increase over the time. The reason for this astonishing behavior becomes clear if one takes into account that $\overline{\sigma_r^{2\,(g)}} = \overline{\sigma_r^{(g)}}^2 + \mathrm{D}^2[\sigma_r]^{(g)}$: As long as σ_r fluctuates with a variance D^2, we have $\overline{\sigma_r^{2\,(g)}} \neq \overline{\sigma_r^{(g)}}^2$. Actually, $\mathrm{D}^2[\sigma_r]^{(g)}$ has influence on the expected population variance $\sigma^2[y]^{(g)}$. That is why we have to consider $\overline{\sigma_r^{2\,(g)}}$ next.

B.2 Determining $\overline{\sigma_r^2}$

The calculation of $\overline{\sigma_r^2}$ starts from (72) and (73), respectively. Taking the square of (72), we obtain for the intermediate recombination

$$\begin{aligned} \sigma_{r_I}^{2\,(g+1)} &= \frac{1}{\mu^2}\sum_{k=1}^{\mu}\sum_{j=1}^{\mu} \sigma_{r_I}^{2\,(g)} \mathrm{e}^{\zeta_k^{(g)}} \mathrm{e}^{\zeta_j^{(g)}} \\ &= \sigma_{r_I}^{2\,(g)} \frac{1}{\mu^2}\sum_{m=1}^{\mu} \mathrm{e}^{2\zeta_m^{(g)}} + \sigma_{r_I}^{2\,(g)} \frac{1}{\mu^2}\sum_{k=1}^{\mu}\sum_{j\neq k} \mathrm{e}^{\zeta_k^{(g)}} \mathrm{e}^{\zeta_j^{(g)}}. \end{aligned} \tag{78}$$

Now the expected value can be calculated taking the statistical independence of the identically distributed ζ_k, ζ_j, and ζ_m into account

$$\begin{aligned} \overline{\sigma_{r_I}^{2\,(g+1)}} &= \overline{\sigma_{r_I}^{2\,(g)}} \left[\frac{1}{\mu^2}\sum_{m=1}^{\mu} \overline{\mathrm{e}^{2\zeta}} + \frac{1}{\mu^2}\sum_{k=1}^{\mu}\sum_{j\neq k} \overline{\mathrm{e}^{\zeta}}^2 \right] \\ &= \overline{\sigma_{r_I}^{2\,(g)}} \left[\frac{1}{\mu}\overline{\mathrm{e}^{2\zeta}} + \frac{\mu^2-\mu}{\mu^2}\overline{\mathrm{e}^{\zeta}}^2 \right]. \end{aligned} \tag{79}$$

Therefore, we get

$$\overline{\sigma_{r_I}^{2\,(g+1)}} = \overline{\sigma_{r_I}^{2\,(g)}} \left(\overline{\mathrm{e}^{\zeta}}^2 + \frac{1}{\mu}\left[\overline{\mathrm{e}^{2\zeta}} - \overline{\mathrm{e}^{\zeta}}^2 \right] \right) \tag{80}$$

or alternatively

$$\overline{\sigma_{r_I}^{2\,(g+1)}} = \overline{\sigma_{r_I}^{2\,(g)}} \left(\overline{\mathrm{e}^{\zeta}}^2 + \frac{1}{\mu}\mathrm{D}^2[\mathrm{e}^{\zeta}] \right) \quad \text{with} \quad \mathrm{D}^2[\mathrm{e}^{\zeta}] = \overline{\mathrm{e}^{2\zeta}} - \overline{\mathrm{e}^{\zeta}}^2, \tag{81}$$

where $\mathrm{D}^2[\mathrm{e}^\zeta]$ is the variance of the mutation operator. Finally, we obtain the equation of the σ_r^2 dynamics

$$\overline{\sigma_{r_I}^{2\,(g)}} = \overline{\sigma_{r_I}^{2\,(0)}} \left(\overline{\mathrm{e}^{\zeta}}^2 + \frac{1}{\mu}\mathrm{D}^2[\mathrm{e}^{\zeta}] \right)^g. \tag{82}$$

One can clearly see $\overline{\sigma_{r_I}^{2\,(g)}} = \overline{\sigma_{r_I}^{(g)}}^2$ does only hold for $\mu \to \infty$ because $\mathrm{D}^2[\mathrm{e}^\zeta] > 0$. As pointed out in the previous subsection, even when $\overline{\mathrm{e}^\zeta} = 1$ were guaranteed (leading to $\overline{\sigma_{r_I}^{(g)}} = \overline{\sigma_{r_I}^{(0)}}$, see Table II), $\overline{\sigma_{r_I}^{2\,(g)}}$ will be an monotonously increasing function.

As to the calculation of $\overline{\sigma_{r_G}^{2\,(g+1)}}$ (geometric recombination) we start from (73)

$$\sigma_{r_G}^{2\,(g+1)} = \left(\prod_{m=1}^{\mu} \sigma_{r_G}^{(g)} \mathrm{e}^{\zeta_m^{(g)}} \right)^{\frac{2}{\mu}} = \sigma_{r_G}^{2\,(g)} \prod_{m=1}^{\mu} \exp\left(\frac{2\zeta_m^{(g)}}{\mu} \right). \tag{83}$$

The expected value reads

$$\overline{\sigma_{r_G}^{2\,(g+1)}} = \overline{\sigma_{r_G}^{2\,(g)}} \prod_{m=1}^{\mu} \overline{\mathrm{e}^{2\zeta/\mu}} = \overline{\sigma_{r_G}^{2\,(g)}} \left(\overline{\mathrm{e}^{2\zeta/\mu}} \right)^{\mu}. \tag{84}$$

Here, we have taken the statistical independence of the identically distributed ζ_m into account. One can rewrite (84) by introducing the variance of $e^{\zeta/\mu}$

$$\mathrm{D}^2[e^{\zeta/\mu}] = \overline{e^{2\zeta/\mu}} - \overline{e^{\zeta/\mu}}^2 \tag{85}$$

leading to

$$\overline{\sigma_{r_G}^{2\,(g+1)}} = \overline{\sigma_{r_G}^{2\,(g)}} \left(\overline{e^{\zeta/\mu}}^2 + \mathrm{D}^2[e^{\zeta/\mu}] \right)^{\mu} . \tag{86}$$

Thus, the evolution equation of $\overline{\sigma_{r_G}^{2\,(g)}}$ becomes

$$\overline{\sigma_{r_G}^{2\,(g)}} = \overline{\sigma_{r_G}^{2\,(0)}} \left(\overline{e^{\zeta/\mu}}^2 + \mathrm{D}^2[e^{\zeta/\mu}] \right)^{\mu g} . \tag{87}$$

One can see that – similar to the intermediate recombination case – a vanishing drift, that is, $\overline{e^{\zeta/\mu}} = 1$ (leading to $\overline{\sigma_{r_G}^{(g)}} = \overline{\sigma_{r_G}^{(0)}}$, see Table II), does not imply $\overline{\sigma_{r_G}^{2\,(g)}} = \overline{\sigma_{r_G}^{2\,(0)}}$. Again, the reason lies in the stochasticity of the σ mutation operator.

B.3 Determination of the expected population variance of the object parameters

The calculation of $\sigma^2[y]^{(g+1)}$ starts from Eq. (62). The first term of (62) is already known from Eq. (63). Therefore, it remains to calculate $\mathrm{E}[\sigma^{2\,(g+1)}]$. Since $\sigma^{(g+1)}$ is generated in the first line of Eq. (61) and Ξ_σ is independent of the σ recombination, we find with (67) and (65/66)

$$\mathrm{E}[\sigma^{2\,(g+1)}] = \overline{e^{2\zeta}}\;\overline{\sigma_r^{2\,(g)}}. \tag{88}$$

Let us consider the intermediate object parameter recombination case first. With (62), (63), and (88) we have (writing g instead of $g+1$)

$$\sigma^2[y]^{(g)} \;=\; 0 \;+\; \overline{e^{2\zeta}}\;\overline{\sigma_r^{2\,(g-1)}} \tag{89}$$

and therefore the evolution dynamics becomes with (82)

$$\underline{(\mu/\mu_{II})\text{-recombination:}} \qquad \sigma^2[y]^{(g)} = \overline{\sigma_r^{2\,(0)}}\;\overline{e^{2\zeta}} \left(\overline{e^{\zeta}}^2 + \frac{1}{\mu}\mathrm{D}^2[e^{\zeta}] \right)^{(g-1)} . \tag{90}$$

for the "II" recombination case.[4] For the "IG" recombination version we obtain with (87)

$$\underline{(\mu/\mu_{IG})\text{-recombination:}} \qquad \sigma^2[y]^{(g)} = \overline{\sigma_r^{2\,(0)}}\;\overline{e^{2\zeta}} \left(\overline{e^{\zeta/\mu}}^2 + \mathrm{D}^2[e^{\zeta/\mu}] \right)^{\mu(g-1)} . \tag{91}$$

As one can see, both the (μ/μ_{II}) and the (μ/μ_{IG}) versions can exhibit exponential variance increase as postulated by Proposition 2 (see Section II-B). As necessary and sufficient condition, the value of the expression in the large parenthesis of (90) and (91), respectively, must be greater than 1. This does even hold when the σ mutation operator is "drift free", that is, $\overline{e^{\zeta}} = 1$ and $\overline{e^{\zeta/\mu}} = 1$, respectively.

The cases with dominant object parameter recombination need further considerations. Instead of (89) we now get from (63), (62), and (88)

$$\sigma^2[y]^{(g)} = \frac{\mu-1}{\mu}\sigma^2[y]^{(g-1)} + \overline{e^{2\zeta}}\;\overline{\sigma_r^{2\,(g-1)}}. \tag{92}$$

Considering (82) and (87), respectively, one sees that (92) is of the form

$$a^{(g)} = b a^{(g-1)} + \left(\frac{c}{d}\right) d^g, \tag{93}$$

with

$$b := \frac{\mu-1}{\mu}, \qquad c := \overline{\sigma_r^{2\,(0)}}\;\overline{e^{2\zeta}} \tag{94}$$

[4] In order to simplify references to different variants of recombination, we will use the notation "OS" where "O" stands for the object parameter recombination types $I|D$ and "S" stands for the strategy parameter recombination types $I|D|G$, that is, "II" means intermediate object parameter recombination and intermediate strategy parameter recombination.

and

$$d_I := \overline{\mathrm{e}^{\zeta}}^2 + \frac{1}{\mu}\mathrm{D}^2[\mathrm{e}^{\zeta}] \tag{95}$$

for the intermediate σ recombination and

$$d_G := \left(\overline{\mathrm{e}^{\zeta/\mu}}^2 + \mathrm{D}^2[\mathrm{e}^{\zeta/\mu}]\right)^{\mu} \tag{96}$$

for the geometric σ recombination.

By recursion, Eq. (93) can be expressed as a sum of geometric progression. It reads

$$a^{(g)} = a^{(0)}b^g + \frac{c}{d}\sum_{k=0}^{g-1} b^k d^{g-k} = a^{(0)}b^g + \frac{cd^g}{d}\sum_{k=0}^{g-1}(b/d)^k. \tag{97}$$

Thus, one gets the closed solution

$$a^{(g)} = a^{(0)}b^g + \frac{c}{d}d^g\frac{1-(b/d)^g}{1-(b/d)} = a^{(0)}b^g + c\frac{d^g - b^g}{d-b}. \tag{98}$$

Finally, we get for (98) with (94) and (95) the population variance dynamics of the "DI" case

$\underline{(\mu/\mu_{DI})\text{-recombination:}}$

$$\sigma^2[y]^{(g)} = \sigma^2[y]^{(0)}\left(1-\frac{1}{\mu}\right)^g + \overline{\sigma_r^{2\,(0)}}\;\overline{\mathrm{e}^{2\zeta}}\;\frac{\left(\overline{\mathrm{e}^{\zeta}}^2 + \frac{1}{\mu}\mathrm{D}^2[\mathrm{e}^{\zeta}]\right)^g - \left(1-\frac{1}{\mu}\right)^g}{\left(\overline{\mathrm{e}^{\zeta}}^2 + \frac{1}{\mu}\mathrm{D}^2[\mathrm{e}^{\zeta}]\right) - \left(1-\frac{1}{\mu}\right)}. \tag{99}$$

For the "DG" recombination version one respectively obtains with (96)

$\underline{(\mu/\mu_{DG})\text{-recombination:}}$

$$\sigma^2[y]^{(g)} = \sigma^2[y]^{(0)}\left(1-\frac{1}{\mu}\right)^g + \overline{\sigma_r^{2\,(0)}}\;\overline{\mathrm{e}^{2\zeta}}\;\frac{\left(\overline{\mathrm{e}^{\zeta/\mu}}^2 + \mathrm{D}^2[\mathrm{e}^{\zeta/\mu}]\right)^{\mu g} - \left(1-\frac{1}{\mu}\right)^g}{\left(\overline{\mathrm{e}^{\zeta/\mu}}^2 + \mathrm{D}^2[\mathrm{e}^{\zeta/\mu}]\right)^{\mu} - \left(1-\frac{1}{\mu}\right)}. \tag{100}$$

In order to use this formulae, the statistical parameters of the mutation operators are needed. For convenience these expressions are collected in Table III.

TABLE III

Statistical parameters of the standard σ mutation operators

	log-normal rule	symmetric two-point rule
$\overline{\mathrm{e}^{\zeta}}$	$\mathrm{e}^{\tau^2/2}$	$\cosh(\varepsilon)$
$\overline{\mathrm{e}^{2\zeta}}$	$\mathrm{e}^{2\tau^2}$	$\cosh(2\varepsilon)$
$\overline{\mathrm{e}^{\zeta/\mu}}$	$\mathrm{e}^{\tau^2/2\mu^2}$	$\cosh(\varepsilon/\mu)$
$\overline{\mathrm{e}^{2\zeta/\mu}}$	$\mathrm{e}^{2\tau^2/\mu^2}$	$\cosh(2\varepsilon/\mu)$
$\mathrm{D}^2[\mathrm{e}^{\zeta}]$	$\mathrm{e}^{\tau^2}(\mathrm{e}^{\tau^2}-1)$	$\cosh(2\varepsilon) - (\cosh(\varepsilon))^2$
$\mathrm{D}^2[\mathrm{e}^{\zeta/\mu}]$	$\mathrm{e}^{\tau^2/\mu^2}(\mathrm{e}^{\tau^2/\mu^2}-1)$	$\cosh(2\varepsilon/\mu) - (\cosh(\varepsilon/\mu))^2$

When comparing the variance evolution of the "II" and "IG" type, Eqn. (90), (91), with those of type "DI" and "DG", Eqn. (99), (100), one sees that the dominant object parameter recombination adds decay terms to the variance evolution. That is, the initial parental population variance fades away exponentially. However, unlike the intermediate (μ/μ) object parameter recombination, there is still a certain "memory" of the older parental states. It is an open question whether this property has some performance benefits in nonlinear fitness landscapes.

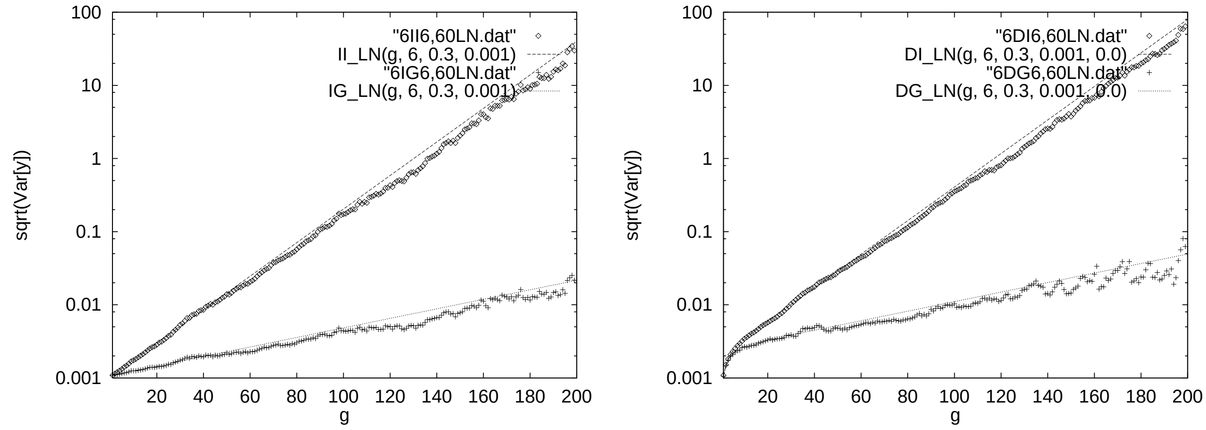

Fig. 6. Evolution of the square root of the expected population variance. Left picture: intermediate object parameter recombination, right picture: dominant object parameter recombination. The simulation results have been obtained by averaging over 500 independent runs. In all experiments $\sigma^2[y]^{(0)} = 0$ and $\sigma_r^{(0)} = 0.001$ has been chosen. As learning parameters $\tau = 0.3$ and $\varepsilon = 0.3$, respectively, have been used.

B.4 Simulation examples and discussion

Figure 6 compares the formulae with simulations on $(6/6, 60)$-ESs using all four variants of recombination and the log-normal mutation operator. One observes a good agreement between theory and experiments.

As one can see, even the geometric recombination variants exhibit the exponential variance increase. However, the time constant is larger. For example, considering (99) and (100) one can estimate that the geometric recombination has an up to μ times larger time constant. This means that the variance evolution slows down by an maximal factor of μ. Choosing μ sufficiently large, one could infer from simulation experiments that the ES versions with geometric recombination are indeed "drift free", as has been argueed by Hansen [19, p. 29]. However, by the analysis presented here, it becomes clear that the variance increasing behavior ("drift" of the σ values) comes from the σ-mutation operators, those standard versions ((67)/(68)) and (70) are already biased toward a σ^2 increase. From this point of view, geometric recombination is more "biased" than intermediate recombination because of its strong μ-dependent damping behavior.

VI. Real-parameter GAs versus Self-adaptive ESs

With the above calculations of expected population variance for real-parameter variation operators and self-adaptive ES operators on flat fitness landscapes, it becomes easier to explain why similar performances of them have been observed earlier [4]. Although the calculations will be different for other fitness landscapes, our postulation allows us to conclude that as long as the expected population variances of a self-adaptive ES and a real-parameter GA are more or less the same, their performance will be similar. For flat fitness landscapes, this allows us to equate the growth of expected population variances of two operators and to find corresponding characteristic parameter values. For example, considering the self-adaptive ES with log-normal update and (μ/μ_{II})-recombination operator and real-parameter GAs with a fuzzy recombination operator, we equate their growth ratio of expected population variances (Eqs. (90) and (59)) to find a relation between characteristic parameter (τ and d) values:

$$\frac{\sigma^2[y]^{(g)}}{\sigma^2[y]^{(g-1)}} = e^{\tau^2} + \frac{1}{\mu_{ES}} e^{\tau^2}\left(e^{\tau^2} - 1\right) = 1 + \frac{d^2}{3} - \frac{1}{\mu_{GA}}. \tag{101}$$

For large population size, the following relationship holds:

$$\tau \approx \sqrt{\ln\left(1 + \frac{d^2}{3}\right)}. \tag{102}$$

Using the above relationship between self-adaptive ESs and real-parameter GAs with FR operator, we would expect to have similar dynamics of the expected population variance on a flat fitness landscape. However, when applied to other fitness landscapes, the expected population variance may not be same. This is because real-parameter GAs and self-adaptive ESs traditionally use different reproduction operators. This makes it difficult to compare them in landscapes other than flat fitness landscapes. As mentioned earlier, reproduction operator has an effect of reducing the population variance in general. Thus, if reproduction operators used in self-adaptive EAs do not have similar property in reducing population variance, comparison of variation operators with identical variance growth properties would be meaningless.

In order to investigate how good this estimate is on the sphere fitness landscapes, we run a self-adaptive ES with $\tau = 0.57$ (calculated using Eq. (101) with $d = 1.095$ and $\mu_{GA} = 100$). To have a similar selection pressure as that in real-parameter GAs, we use binary tournament selection operator, instead of an usual truncation selection in the self-adaptive ES. Thus, we use $\mu_{ES} = \rho = 50$, but use a children population size $\lambda = 100$. Simulation results with (μ/μ_{II})-recombination operator are shown in figures 4 and 5 (average of 10 runs). The performance of self-adaptive ESs are not significantly different from that of real-parameter GAs. The difference in the performance arises due to use of estimates for flat fitness landscapes in a sphere model. It has been observed that for $\tau \approx 0.4$, the self-adaptive ESs show very similar linear order convergence as demonstrated by real-parameter GAs.

It is interesting to note that the expected population variance *never* reduces with self-adaptive ESs. This is due to the way self-adaptive mutation operators are constructed. On the other hand, the expected population variance in real-parameter GAs can increase or decrease depending on the population size. If the population size is small, a decrease in expected population variance is likely. Thus, there is a greater need of choosing an appropriate population size in GAs than in self-adaptive ESs.

Clearly, the calculations of expected population variance and their relavance in the performance of an algorithm demonstrated here mark a first step towards finding a better connection among self-adaptive evolutionary algorithms.

VII. On the current state of the analysis of EAs in real-valued linear fitness landscapes

Proposition 3 of our postulates in Section II-B concerns the behavior in linear fitness landscapes. Up until now, the analysis of self-adaptive properties in EAs on linear fitness landscapes has not been done. As an exception, the analysis of the $(1,\lambda)$-σSA-ES on the sphere model was presented in [15]. Although not explicitly considered there, the linear fitness case is implicitly included in that analysis. This is so because, given fixed mutation strength σ, the normalized mutation strength σ^* defined by

$$\sigma^* := \sigma \frac{N}{R}, \tag{103}$$

with R as the actual parental distance to the optimum and N the object parameter space dimension, goes to zero when $R \to \infty$. However, this means that with increasing R the spherical landscape appears for the mutations with strength σ more and more plane. In the limit one gets the behavior of a hyperplane (linear fitness landscape) if $\sigma^* \to 0$. That is, by considering the limit $\sigma^* \to 0$ for $0 < \sigma < \infty$, the result of the sphere model theory can be used to describe the behavior of the ES in linear fitness landscapes.

A. Determining the expected σ evolution

The central quantity of the σSA theory is the so-called *self-adaptation response function* $\psi(\sigma^{*(g)})$. It is defined for the $(1,\lambda)$-σSA-ES as

$$\psi(\sigma^{*(g)}) := \mathrm{E}\left[\frac{\sigma^{(g+1)} - \sigma^{(g)}}{\sigma^{(g)}}\right] \tag{104}$$

and measures the relative change of the strategy parameter σ given a parental state σ and σ^*, respectively. Figure 7 shows typical graphs of $\psi(\sigma^{*(g)})$ functions. From this curves only the value at $\sigma^* = 0$ is of interest for linear fitness landscapes. Unfortunately, in almost all cases there is no analytical expression for $\psi(\sigma^*)$. The only exception is for the $(1,2)$-σSA-ES with two-point mutation operator (70). One finds in this special case [22, p. 27]

$$\text{symmetrical two-point rule:} \qquad \psi_{1,2}(0) = \frac{1}{2}\frac{(\alpha-1)^2}{\alpha} > 0. \tag{105}$$

For small learning parameters τ and $\alpha - 1$, respectively, it is possible to derive approximate $\psi(\sigma^*)$ expressions [15, p. 328]

$$\text{log-normal rule:} \qquad \psi_{1,\lambda}(0) = \tau^2\left(d^{(2)}_{1,\lambda} - \frac{1}{2}\right) + \ldots > 0, \tag{106}$$

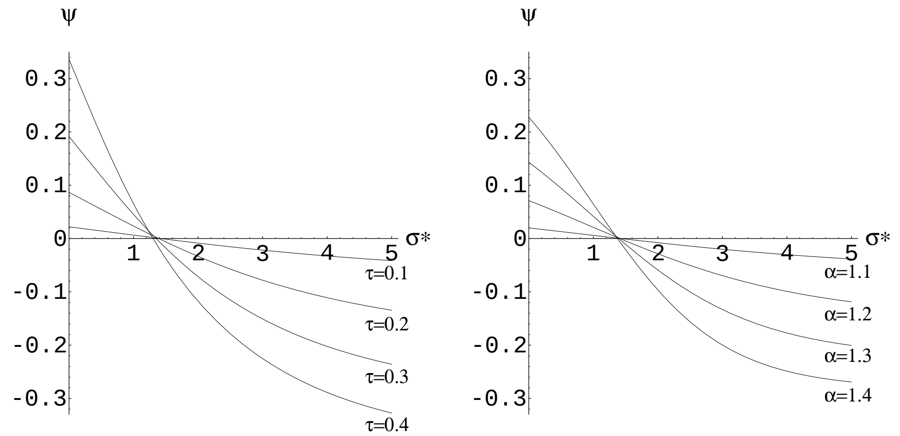

Fig. 7. The self-adaptation response $\psi(\sigma^*)$ for the $(1,10)$-σSA-ES with $N = 30$. Figure left: log-normal rule (67/68) used. Figure right: symmetric two-point rule (70) used.

$$\text{symmetrical two-point rule:} \qquad \psi_{1,\lambda}(0) = (2-\alpha)(\alpha-1)^2\left(d^{(2)}_{1,\lambda} - \frac{1}{2}\right) + \ldots > 0, \tag{107}$$

with $d^{(2)}_{1,\lambda}$ the second order progress coefficient (see e.g. [15, p. 325]).

In order to calculate the σ dynamics on flat fitness landscapes Eq. (104) is used to express the expected σ at generation $g+1$ by that of g. One immediately finds with $\sigma^* = 0$

$$\overline{\sigma^{(g+1)}} = (1 + \psi(0))\,\overline{\sigma^{(g)}}. \tag{108}$$

Therefore, the dynamics of the expected σ evolution reads

$$\overline{\sigma^{(g)}} = \sigma^{(0)}(1+\psi(0))^g. \tag{109}$$

Since $\psi(0) > 0$ for the σ mutation rules considered here, one observes an exponentially increase of the expected mutation strength. As a consequence, the population variance of the object parameters must increase as well. This is again regarded as a desired behavior. In the doctoral thesis of Ostermeier [12, p. 11] it was claimed that a $(1,2)$-σSA-ES does not exhibit such a behavior. Since he uses the symmetric two-point rule, we are able to disprove his assertion rigorously: Because of the exact result (105), the $(1,2)$-σSA-ES does necessarily increase the expected σ according to (108) as long as $\alpha > 1$. The error in Ostermeier's argumentation comes again from the consideration of improvement probabilities. It could be avoided by looking at the expected σ changes as has been done here.

B. Determining the expected distance traveled

Unlike the cases considered in the previous sections, the calculation of the parental variance does not make sense for the case of $(1,\lambda)$ strategies considered here (note, there is only one parent). One could consider the offspring variance instead. However, alternatively we can investigate the expected distance traveled in optimum (gradient) direction.

Assuming small learning parameters, the expected distance $\mathrm{E}[\Delta x]$ traveled from generation g to $(g+1)$ is roughly

$$\mathrm{E}[\Delta x]^{(g)} = \overline{x^{(g)}} - \overline{x^{(g-1)}} = c_{1,\lambda}\overline{\sigma^{(g)}} + \ldots, \tag{110}$$

with the progress coefficient[5] $c_{1,\lambda}$ (for a table of values, see e.g. [15, p. 325]). Formula (110) is exact for small learning parameters (that is, $\tau \to 0$, $\alpha \to 1$). However, it may be well used as a first approximation here. From (110) we get with (109)

$$\mathrm{E}[\Delta x]^{(g)} = c_{1,\lambda}\sigma^{(0)}(1+\psi(0))^g + \ldots \tag{111}$$

[5]Eq. (110) holds exactly for isotropic normally distributed mutations (that is, for $\tau \to 0$ or $\alpha \to 1$). In this case, the problem reduces to a one-dimensional: Starting from a point $x^{(g-1)}$, λ offspring are generated by $\sigma\mathcal{N}(0,1)$ mutations z_l. The largest mutation, denoted by $z_{\lambda:\lambda}$ produces the best offspring and therefore the parent $x^{(g)}$ for the next generation. The variate $z_{\lambda:\lambda}$ is known as the λth order statistics (see e.g. [23]) of the $\mathcal{N}(0,\sigma^2)$ variate, and its expectation is $c_{1,\lambda}\sigma$, with the progress coefficient $c_{1,\lambda}$.

and therefore, by assuming an initial $x^{(0)} = 0$, we get

$$\overline{x^{(g)}} = \overline{x^{(g-1)}} + c_{1,\lambda}\sigma^{(0)}(1+\psi(0))^g + \dots. \tag{112}$$

By recurrence one obtains a sum of geometric progression

$$\begin{aligned} \overline{x^{(g)}} &= c_{1,\lambda}\sigma^{(0)} \sum_{k=1}^{g} (1+\psi(0))^k + \dots \\ &= c_{1,\lambda}\sigma^{(0)} \left(-1 + (1+\psi(0))^g + \sum_{k=0}^{g-1} (1+\psi(0))^k \right) + \dots \\ &= c_{1,\lambda}\sigma^{(0)} \left(-1 + (1+\psi(0))^g + \frac{1-(1+\psi(0))^g}{1-(1+\psi(0))} \right) + \dots. \end{aligned} \tag{113}$$

After rewriting (113) we end up with

$$\overline{x^{(g)}} = c_{1,\lambda}\sigma^{(0)} \left(1 + \frac{1}{\psi(0)} \right) [(1+\psi(0))^g - 1] + \dots. \tag{114}$$

As one can see, for sufficiently large g the self-adaptation algorithm exhibits an exponential increase of the distance traveled in optimum direction. As an example, the $(1,10)$-σSA-ES is simulated and the graph of (114) are plotted in Figure 8. One observes a good agreement over may orders of magnitude.

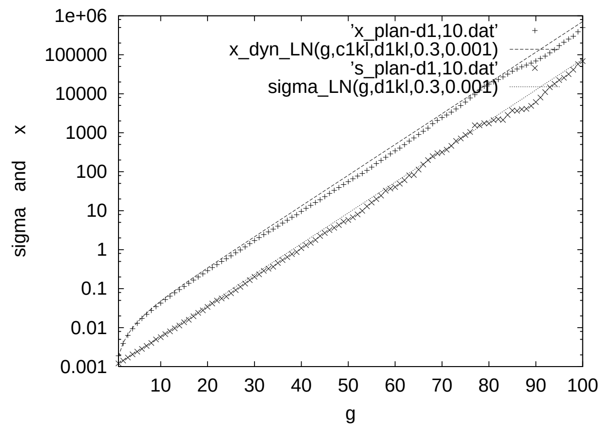

Fig. 8. The σ-evolution and the distance x traveled toward the optimum produced by a standard $(1,10)$-σSA-ES using log-normal mutations on a linear fitness landscape. As learning parameter $\tau = 0.3$ has been chosen. The simulation results have been obtained by averaging over 500 independent runs. The parameter space dimension (search space) was $N = 30$.

VIII. Conclusions

In this paper, we postulate the properties on population mean and variance for self-adaptive evolutionary algorithms (SA-EAs). With the task of exploration and exploitation in mind, we argue that the population mean should not be changed by a variance operator. Although the manner in which the population variance should be changed by a variance operator must depend on the fitness landscape and the associated reproduction operator, we argue that it may be better strategy, in general, to have a certain tendency of increasing the population variance by a variation operator (that is, by recombination and mutation).

We have analyzed the population mean and variance of three crossover operators—blend crossover (BLX), simulated binary crossover (SBX), and fuzzy recombination (FR)—commonly-used with real-coded GAs. Theoretical predictions of increasing and decreasing nature of population variance are validated with experimental results on a flat fitness landscape. By choosing an appropriate characteristic parameter for each crossover operator so as to make the population variance the same, we have shown that GAs with all three crossover operators exhibit similar self-adaptive feature on a sphere model.

For the first time, the population variance has been calculated for self-adaptive evolution strategies (ESs). Theoretical results have also been supported by showing simulation results on a flat fitness function. An estimate of the expected distance traveled on a linear fitness function has also been calculated.

Using the estimates for expected population growth on a flat fitness landscape, the performance of self-adaptive ESs and real-parameter GAs are compared on a sphere model. Although the results have shown similar behaviors, a closer similarity can be achieved if the expected population variation estimates can be obtained for both reproduction and variation operators.

Acknowledgments

The first author acknowledges support as Heisenberg Fellow of the Deutsche Forschungsgemeinschaft under grant Be1578/4-1. The second author acknowledges the support from Alexander von Humboldt Foundation, Germany and Department of Science and Technology, India during the course of this study.

References

[1] Hans-Paul Schwefel. Collective phenomena in evolutionary systems. In P. Checkland and I. Kiss, editors, *Problems of Constancy and Change — the Complementarity of Systems Approaches to Complexity, Papers presented at the 31st Annual Meeting of the Int'l Soc. for General System Research*, volume 2, pages 1025–1033, Budapest, 1.–5. Juni 1987. Int'l Soc. for General System Research.

[2] L. J. Fogel, P. J. Angeline, and D. B. Fogel. An evolutionary programming approach to self-adaptation on finite state machines. In J. R. McDonnell, R. G. Reynolds, and D. B. Fogel, editors, Proceedings of the Fourth International Conference on Evolutionary Programming, pages 355–365, , 1995.

[3] N. Hansen, A. Ostermeier, and A. Gawelczyk. On the Adaptation of Arbitrary Normal Mutation Distributions in Evolution Strategies: The Generating Set Adaptation. In L. J. Eshelman, editor, *Proc. 6th Int'l Conf. on Genetic Algorithms*, pages 57–64, San Francisco, CA, 1995. Morgan Kaufmann Publishers, Inc.

[4] K. Deb and H.-G. Beyer. Self-adaptive Genetic Algorithms with Simulated Binary Crossover. Technical report no. CI-61/99, University of Dortmund, Department of Computer Science/LS11, 1999.

[5] H. Kita. A comparison study on self-adaptation in evolution strategies and real-coded genetic algorithms. Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, Japan, 1998.

[6] H.-M. Voigt, H. Mühlenbein, and D. Cvetković. Fuzzy recombination for the Breeder Genetic Algorithm. In L. J. Eshelman, editor, *Proc. 6th Int'l Conf. on Genetic Algorithms*, pages 104–111, San Francisco, CA, 1995. Morgan Kaufmann Publishers, Inc.

[7] L. J. Eshelman and J. D. Schaffer. Real-coded genetic algorithms and interval schemata. In L. D. Whitley, editor, *Foundations of Genetic Algorithms, 2*, pages 187–202. Morgan Kaufmann, San Mateo, CA, 1993.

[8] K. Deb and R. B. Agrawal. Simulated binary crossover for continuous search space. *Complex Systems*, 9:115–148, 1995.

[9] I. Ono and S. Kobayashi. A real-coded genetic algorithm for function optimization using unimodal normal distribution crossover. In T. Bäck, editor, Proceedings of the Seventh International Conference on Genetic Algorithms, pages 246–253, Morgan Kauffman, San Mateo, CA, 1997.

[10] K. Deb and H.-G. Beyer. Self-Adaptation in Real-Parameter Genetic Algorithms with Simulated Binary Crossover. In W. Banzhaf, J. Daida, A.E. Eiben, M.H. Garzon, V. Honavar, M. Jakiela, and R.E. Smith, editors, *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference*, San Francisco, CA, 1999. Morgan Kaufmann.

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

[12] A. Ostermeier. *Schrittweitenadaptation in der Evolutionsstrategie mit einem entstochastisierten Ansatz*. Doctoral thesis, Technical University of Berlin, Berlin, 1997.

[13] K. Deb. *Optimization in engineering design: Algorithms and examples*. Prentice-Hall, New Delhi, 1995.

[14] T. Bäck, D. Fogel, and Z. Michalewicz, editors. *Handbook of Evolutionary Computation*, New York, 1997. IOP Publishing and Oxford University Press.

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

[16] H.-G. Beyer. On the Dynamics of EAs without Selection. In W. Banzhaf and C. Reeves, editors, *Foundations of Genetic Algorithms, 5*, pages 5–26, San Mateo, CA, 1999. Morgan Kaufmann.

[17] L. J. Eshelman. The CHC Adaptive Search Algorithm: How to Have Save Search When Engaging in Nontraditional Genetic Recombination. In G. J. E. Rawlins, editor, *Foundations of Genetic Algorithms, 1*, pages 265–283. Morgan Kaufmann, San Mateo, CA, 1991.

[18] H.-G. Beyer. Toward a Theory of Evolution Strategies: On the Benefit of Sex – the $(\mu/\mu, \lambda)$-Theory. *Evolutionary Computation*, 3(1):81–111, 1995.

[19] N. Hansen. *Verallgemeinerte individuelle Schrittweitenregelung in der Evolutionsstrategie*. Doctoral thesis, Technical University of Berlin, Berlin, 1998.

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

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

[22] H.-G. Beyer. Towards a Theory of 'Evolution Strategies': The $(1, \lambda)$-Self-Adaptation. Technical Report SYS-1/95, Department of Computer Science, University of Dortmund, 1995.

[23] B. C. Arnold, N. Balakrishnan, and H. N. Nagaraja. *A First Course in Order Statistics*. Wiley, New York, 1992.