Evolutionary Algorithms in Noisy Environments: Theoretical Issues and Guidelines for Practice

Hans-Georg Beyer

University of Dortmund, Department of Computer Science XI
D-44221 Dortmund, Germany
beyer@zappa.cs.uni-dortmund.de

Abstract

This paper is devoted to the effects of fitness noise in EAs (evolutionary algorithms). After a short introduction to the history of this research field, the performance of GAs (genetic algorithms) and ESs (evolution strategies) on the hyper-sphere test function is evaluated. It will be shown that the main effects of noise – the decrease of convergence velocity and the residual location error R_∞ – are observed in both GAs and ESs.

Different methods for improving the performance are presented and hypotheses on their working mechanisms are discussed. The method of rescaled mutations is analyzed in depth for the $(1, \lambda)$-ES on the sphere model. It is shown that this method needs advanced self-adaptation techniques in order to take advantage of the theoretically predicted performance gain. The troubles with current self-adaptation techniques are discussed and directions for further research will be worked out.

Key words: evolutionary algorithms (GA, ES, EP), noisy fitness data, convergence properties, optimization under noise, convergence improvement techniques

1 Introduction

A supposed advantage of evolutionary algorithms (EAs) is that EAs are believed to work well in noisy environments. This is in contrast to traditional optimization methods which strongly rely on deterministic information in order to find optimal solutions. This "believed" convergence stability of EAs rests on at least two observations: First, evolution in Nature appears to be highly disturbed by disinformation, deception, and noise. Nevertheless, the living beings seem to be well adapted to their environment. Provided that

Darwinian evolution can be regarded as optimization, or at least as some kind of melioration, algorithms designed in accord to Darwin's paradigm should obey similar properties. Second, many EA applications have indeed to cope with noisy fitness information. And it seems that they work well and that noise can even be helpful in evolutionary search.

Although there are some thousand papers published in the field of evolutionary algorithms (EAs), there are only a few ten explicitly dedicated to the problem of noisy fitness measurements. This may appear as a surprise for the reader because in practice one has often to cope with statistical measurement errors. For example, when trying to optimize the operation of a machine tool by tuning machine control parameters, the outcome produced will not be identical even though all parameters have been fixed. Another example concerns the field of computer simulations where numerical errors or even the simulation technique itself (i.e. Monte Carlo simulations, discrete event simulations) may produce "noisy" results. In all such cases improving the outcome can be a risky endeavor because one can never be sure that a seeming improvement obtained by a certain control parameter change is a *real* improvement. That is, the noise may deceive the decision making. It is quite clear that the "degree of deception" must be a function of the *relative* noise level. If the noise strength is small compared to the effective signal strength, deception should be a relatively seldom event. However, how small must be "small", and what is to be expected when the noise level is gradually increased?

The extreme case of a large noise level (large, compared to the effective signal strength) can be easily treated: There is no useful selection information, i.e., a measured improvement can be a result of a (desirable) control parameter change as well as a result of the noise with equal likelihood. Since parameter changes in EAs are usually unbiased and at random, the parameters (to be optimized) will perform a random walk. That is, there will be no directed evolution. A diffusion-like behavior will be observed instead, with the result that the distance to the optimum increases with the time.

Between the two extremes "zero" noise and "total" noise there is the working domain of living beings and of the EAs. It is one of the goals of this paper to investigate the influence of noise on the performance of genetic algorithms (GAs) and evolution strategies (ESs). Hereby, we want to emphasize the *similarities* in the performance behavior of these – at first glance – very different algorithms. To this end, it is expedient to consider very simple fitness functions (the objective function to be optimized by the EA) in order not to be dissuaded from the universal phenomenon by pecularities of the functions to be optimized. Furthermore, simple fitness functions, like e.g. the sphere model considered here, can allow for an *analytical* treatment of the performance behavior of the algorithms. This has the advantage that one can compare the theoretical predictions with the performance of the *real* algorithm. As we will see later on, discrepancies between theory and practice may serve as a driving force for further improvements of the algorithms developed.

The rest of this paper is organized as follows. The next section presents a short history of works dealing explicitly with performance investigations in EAs with noisy fitness data. Section 3 is devoted to the effects of noise on a simple test function – the sphere model. The performance of several GAs and ESs will be compared providing some astonishing results. Section 4 summarizes techniques for convergence improvement and introduces the method of *rescaled* mutations. After having a more general viewpoint up to here, the following sections are devoted to ESs only. In Section 5 the technique of rescaled mutations is analyzed for the so-called $(1, \lambda)$-ES in the frame of the finite-dimensional sphere model. Though the analysis will show that the method can improve the convergence toward the optimum in real-valued search spaces, it proves difficult to achieve this in real algorithms because it affords the control of the standard deviation of the isotropic mutations. The usual way to do this in ESs is by *self-adaptation* (SA). However, as to the rescaled mutation technique, the SA has considerable problems to tune the mutation strength such that the algorithm can benefit from the rescaling effect. Therefore, Section 6 will provide advanced SA techniques that seem to be able to drive the algorithm into its optimal working regime. After that, the paper will close with a short summary and outlook in Section 7.

2 A short history on noise related work

Since high noise levels are observed in nature and because of the hypothesis that Darwinian evolution is optimization one infers that EAs should be highly noise resistant. Unfortunately, up until now (June, 1998) there are only a few serious investigations in this field, even though – probably the first – dates back to the early 70s. It was Rechenberg [1] who tried to evaluate the ES (evolution strategy) by theoretical performance measures on simple fitness functions, as e.g. the N-dimensional sphere (for its definition, see below) and the N-dimensional corridor. For the latter one he succeeded in calculating the progress rate of the noisy $(1+1)$-ES. [1] He found that the expected progress φ in corridor direction decreases toward zero with increasing noise strength (for respective definitions, see below). As to the sphere model, we had to wait yet

[1] The notation $(\mu + \lambda)$ and (μ, λ) refer to the kind of selection used in ESs: μ parents generate λ offspring. The parents for the next generation are obtained by selecting the μ best offspring in case of the (μ, λ) version. In the $(\mu + \lambda)$ version, the parents are obtained from *both* the older generation and the offspring. That is, in comma strategies parents die out per definition, whereas in plus strategies parents can survive infinitely long (so-called "elitist selection").

another twenty years [2].

In 1988, Fitzpatrick and Grefenstette [3] published their empirical results on noisy fitness evaluations in GAs (genetic algorithms). One obvious way of improving the GA performance is through noise reduction by *resampling*, i.e. averaging over a number m of fitness measurements (keeping the control parameters to be optimized constant), however, this increases the number of fitness evaluations by a factor of m. Alternatively, one might also increase the population size λ. Given a fixed amount of CPU-time or total number of fitness evaluations ν, the question arises how to allocate these resources to m and λ in order to get maximal performance. Fitzpatrick and Grefenstette found that the best performance on a biquadratic test function was achieved for small sample sizes m, but large population sizes λ. Is this the general rule?

At least for $(1 \overset{+}{,} \lambda)$-ESs (evolution strategies), on the sphere model, Beyer (1993) [2] has shown by solving the pending progress rate problem that the opposite case can be possible. His predictions initiated further empirical research in the ES field by Hammel and Bäck (1994) [4,5], confirming Beyer's progress rate theory. More interestingly, they found evidences that the preference of resampling over enlarging the population size does even hold for recombinant ESs. However, due to the lack of theory, their results should be taken with some care. As we will see later on, the performance of the real ES can strongly depend on the correct working of the SA (self-adaptation) mechanism controlling the mutation strength of the ES.

Concerning the influence of the SA on the performance, similar observations have been made by Angeline (1996) [6] who compared self-adaptive evolutionary programming (see e.g. Fogel [7]) using a Gaussian mutation rule for the mutation strength with the log-normal mutation rule usually preferred in ESs (for its definition, see Point 3.2.2, Eq. (9)). He reported that the Gaussian rule outperforms the log-normal rule on a set of test functions. Although these performance differences are significant, given the experimental conditions, they appear not to be of orders of magnitude. Again, we have a lack of theory to understand the observations made.

Another observation made in [5] indicates that the reliability of convergence to the global optimum can be improved by a certain amount of noise. Similar observations have been made by Levitan and Kauffman (1995) [8], who investigated a $(1 + 1)$-ES-like "adaptive walk algorithm". However, the latter authors used a bit-mutation operator which performs one-bit moves only. Their "peaks-melting-off effect", i.e. the leaving of local attractors, can as well be accomplished by simulated annealing like selection or by a mutation operator allowing for moves with Hamming distance larger than one. Rana et al. (1996) [9] came to similar assessments and they found that added noise can be helpful for some fitness functions during the initial phases of search.

Theoretical work related to noise in the field of GAs can be traced back at least to the early 90s, when Goldberg and Rudnick [10] developed extensions to the schema theorem [11] in order to account for the sampling noise (so-called collateral noise). These investigations were mainly intended to derive models for the population sizing problem. Population sizing under the presence of fitness noise has been considered in Goldberg et al. [12] first; a more refined model has been given in Harik et al. [13]. The basic idea of these models comes from the paradigm of building block assembly: In order to obtain the final optimum solution, the right building blocks must flow together. The decision process of the GA for selecting the right building blocks, however, is disturbed by the finite population size sampling and the schema fitness variance (collateral noise). By calculating the probability of selecting the right building blocks, depending on the population size, one has an approach for estimating the population size. Fitness noise can be regarded as an additional independent noise source, and thus, it can be easily incorporated into this population sizing approach. Although this approach appears totally different to that which is based on progress analysis in ESs, some of the resulting equations are similar with respect to their functional structure as we will see in Point 3.2.2.

Performance analyses (defined in the sense of this paper) of GAs with noisy fitness evaluations are relatively new. They date back to 1997 where Miller [14] firstly presents in his PhD thesis a usable population sizing model which includes fitness noise. An overview of some of his main results can be found in Miller and Goldberg [15]. Unlike the aforementioned work [10,12,13], it is more in the spirit of ES performance analysis, treating the GA as a dynamical system by analyzing the expected fitness change over the time. In [15] the analysis mainly concentrats on the fitness dynamics of the OneMax bit-counting function which might be regarded as the counterpart of the sphere model in binary search spaces. Due to the similarities in the theoretical approaches we can expect similarities and we will compare with results of this paper. However, there are also differences, e.g. the approximations used neglect the effects of sampling in finite populations, leading Miller to the assertion that fitness noise does not affect proportionate selection [14, p. 75]. As we will see by simple simulations (see Fig. 1 in Section 2), this is only conditionally correct.

Finite sample size effects are addressed in the Ph.D. work of Rattray [16], published in Rattray and Shapiro (1997) [17], who applied their theory to the OneMax bit-counting function and a perceptron learning problem with binary weights using a GA with Boltzmann selection (a special kind of nonlinear proportionate selection). One of their main messages is that the effect of noise can be "removed" by choosing the population size λ according to $\lambda_0 \exp(\beta^2 \sigma_\varepsilon^2)$, where λ_0 stands for the population size of the GA without noise, β is the Boltzmann selection strength, and σ_ε is the standard deviation of the Gaussian fitness noise. It should be mentioned that a similar relation does hold for the

$(1, \lambda)$-ES. This gives us another clue that the effects of noise and the associated problems might be similar in all EAs (evolutionary algorithms). Therefore, it is one of the goals of this paper to search for similarities in the behavior of GAs and EAs with respect to the influence of fitness noise on the optimization performance. The next section exactly serves this goal.

3 On the effect of fitness noise in EAs

It is a very difficult task to evaluate the performance of EAs on real-world problems by theoretical means. Due to their intrinsic probabilistic behavior, it is usually excluded to calculate the dynamics of the optimization process by exact *and* analytical (not numerical) Markov models. Approximations and asymptotic correct approaches are to be used instead. But, even the derivation of approximations has to rely on simple objective functions. One might regard this as a flaw of current EA theories, however, it is most likely that progress in EA theory and in the understanding of the working of these algorithms is obtained by starting from simple models and gradually proceed to more complicated ones.

In this section, the optimization performance of ESs and GAs on the simple N-dimensional sphere model will be evaluated by experiments. We will investigate the influence of different fitness noise levels on the mean value dynamics of the residual distance R of the population to the (known) optimum solution in the parameter space. After introducing the test fitness function to be optimized, Point 3.2.1 is devoted to standard genetic algorithms. Whereas, Point 3.2.2 applies the same tests to different variants of evolution strategies. As a surprise, we will see that the qualitative behavior of both GAs and ESs are similar. This observation will further motivate us to look for similarities in theoretical ES/GA results found so far.

3.1 Fitness and noise models

In order to get a feeling how fitness noise degrades the EA performance, we will consider a simple fitness model to be optimized which leaves a certain chance to succeed with a theoretical analysis. In the GA field, the OneMax bit-counting function [18] is often used for this purpose. In the ES field the N-dimensional sphere model is the most prominent test function defined for real-valued search spaces. In the sequel we will concentrate on the sphere. This decision is based on the fact that there is already a well developed theory for noisy $(1 \overset{+}{,} \lambda)$-ESs [2].

Let $\mathbf{y} = (y_1, \ldots, y_N)^{\mathrm{T}} \in \mathbb{R}^N$ be the parameter vector to be optimized and $\hat{\mathbf{y}}$ the optimal parameter vector, then the general sphere model fitness function reads

$$\tilde{F}(\mathbf{y}) := f(\|\mathbf{y} - \hat{\mathbf{y}}\|) + \varepsilon(\|\mathbf{y} - \hat{\mathbf{y}}\|). \tag{1}$$

Here, $f(r)$ is (usually) a monotonic function with $r : \|\mathbf{y} - \hat{\mathbf{y}}\| \geq 0$, and ε is a Gaussian noise term with zero mean and standard deviation σ_ε

$$\varepsilon = \mathcal{N}(0, \sigma_\varepsilon^2(r)). \tag{2}$$

Assuming a normal noise distribution may be regarded as an approximation of reality based on the maximum entropy principle [19]. Furthermore, it will simplify the derviations considerably. The pdf (probability density function) of the noise reads

$$p(\varepsilon) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \exp\left[-\frac{1}{2}\left(\frac{\varepsilon}{\sigma_\varepsilon}\right)^2\right]. \tag{3}$$

Note, the noise strength σ_ε can be a function of r, allowing for the modeling of *relative* measuring errors.

We will evaluate the performance of sGAs (standard GAs) and sESs (standard ESs) on the special sphere model

$$F(\mathbf{y}) := 1 - \sum_{i=1}^{N} y_i^2 + \mathcal{N}(0, \sigma_\varepsilon^2), \qquad \sigma_\varepsilon^2 = \text{const.} \tag{4}$$

Later on we will use the special fitness model

$$\tilde{Q}(\mathbf{y}) := c \cdot \left(\sum_{i=1}^{N} y_i^2\right)^{\frac{\alpha}{2}} + \mathcal{N}(0, \sigma_\varepsilon^2), \qquad c > 0, \quad \alpha > 0 \tag{5}$$

to derive a N-dependent progress rate formula for the ES with rescaled mutations. Both fitness models have their optimum at $\mathbf{y} = \hat{\mathbf{y}} = (0, \ldots, 0)^{\mathrm{T}}$ when the noise is switched off. Model (4) is for maximization, $\max[F] = 1$, and Model (5) is for minimization, $\min[Q] = 0$ (the symbol Q is used instead of $\tilde{Q}$ when the noise is switched off).

3.2 Performance of sGAs and sESs on the noisy sphere model

3.2.1 GA performance

For the experiments, Eq. (4) serves as fitness model. In the standard GA (Goldberg, 1989) [11], binary coding has been used with 10 bits per parameter y_i and an individual string length $\ell = 100$ leading to a parameter space dimension $N = 10$. The coding was done in such a way that the optimum point $\hat{\mathbf{y}} = \mathbf{0}$ can be exactly expressed by the bit string in order to avoid genotype-phenotype approximation errors. As search interval $y_i \in \left[-\frac{1}{\sqrt{10}}, \frac{1}{\sqrt{10}}\right]$ was chosen with the intention to ensure $0 \leq F(\mathbf{y}) \leq 1$ for $\sigma_\varepsilon = 0$. Recombination was done by uniform crossover (Syswerda, 1989 [20]) with probability $p_{\mathrm{co}} = 1$. The following selection method are investigated: proportionate selection, realized by roulette wheel selection (Goldberg, 1989) [11], and tournament selection with tournament size tourn = 2 (so-called binary tournaments) and tournament size tourn = 5. All selection methods are realized with replacement, i.e., the selected parents are put back into the parental pool after reproduction. The mutation rate p_{m} was set to zero, $p_{\mathrm{m}} = 0$, therefore a random initialization of the population is required. Figure 1 shows the dynamics of the GAs. The plots are average results obtained over 50 independent evolution runs. The left pictures are for a population size (pop_size) $\lambda = 60$, whereas the right pictures display the $\lambda = 120$ case. In each picture, the population average of the fitness $\langle F \rangle$ and of the *residual distance* $\langle R \rangle$ to the optimum in the N-dimensional parameter space is plotted. The upper four curves belong to $\langle F \rangle$ and the lower four curves are for $\langle R \rangle$ showing the influence of four different noise strengths (s_e) $\sigma_\varepsilon = 0$, 0.1, 0.3, and 1.0.

Displaying the residual distance $\langle R \rangle$ is somewhat unusual for GA performance investigations; however, it delivers the additional information how the population approaches the optimum in the N-dimensional (phenotype) parameter space. As one can infer from the pictures, $\langle R \rangle$ approaches a steady state value $\langle R \rangle \to R_\infty$. Obviously, the steady state value R_∞ is a monotonically increasing function of the noise strength σ_ε. For $\sigma_\varepsilon > 0$, there remains always a certain distance to the optimum. The population cannot get nearer to the optimum.

Another interesting observation concerns the selection types used. Proportionate selection appears to be the most *insensitive* selection technique as to the influence of σ_ε on R_∞. For sufficiently large σ_ε, it performs equally well or even better than other selection techniques. However, unlike Miller's statement [14, p. 75], the steady state value R_∞ depends on the noise strength σ_ε. For $\sigma_\varepsilon = 0$, 0.1, and 0.3 the R_∞ curves in the upper pictures of Figure 1 are almost identical, however, for $\sigma_\varepsilon = 1.0$ one observes a separation after a certain number of generations. When the population size λ is increased, this separation is shifted to larger generation numbers. As one can easily show, the

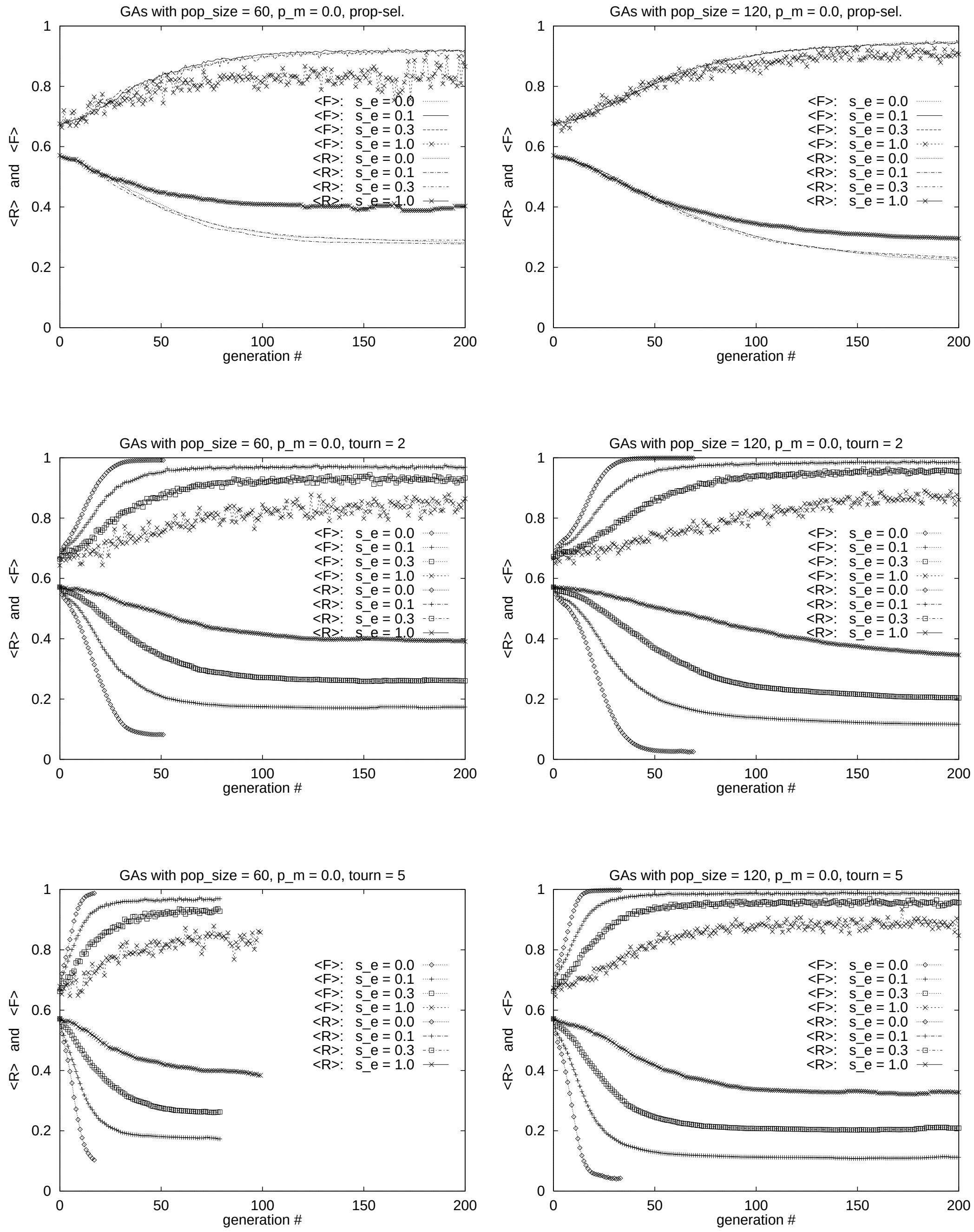

Fig. 1. GA dynamics of the $\langle F\rangle$ and $\langle R\rangle$ averages for four noise levels s_e and different population sizes (left pictures: $\lambda = 60$, right pictures: $\lambda = 120$). In the upper two pictures the case of proportionate selection is displayed, the middle two are obtained from binary tournament selection, and the bottom pictures are from tournament selection with tourn = 5. Note, some of the curves are ending before generation # 200. This is due to gene convergence. This (premature) convergence time can be increased by increase of the population size.

influence of noise can be totally removed for $\lambda \to \infty$. However, this behavior is not a peculiarity of proportionate selection. It holds for tournament and (μ, λ) truncation selection, too, as one can see in the lower four pictures of

Figure 1 and also in Figure 2 to be discussed below. Apart from the population size, the selection pressure has an important influence on the dynamics. As a rule of thumb, increasing the tournament size from two to larger values usually reduces the residual distance R_∞ and speeds up the convergence velocity. This is brought, however, at the price of faster premature gene convergence (see lower pictures in Figure 1). The gene convergence can be avoided by a mutation rate $p_{\mathrm{m}} > 0$, but it increases the R_∞ value, too (observed in simulations not presented here). The premature gene convergence can be shifted to higher generation numbers by increasing the population size λ, as can clearly be seen by comparing the lower two pictures of Figure 1. Furthermore, it is important to note that elitism (keeping track of the best solution found so far) is not very helpful in this scenario because the fitness of the seemingly best individual may be a result of a large noise fluctuation (see the discussion of the ES, below).

3.2.2 ES performance

It is often claimed that ESs are specially tailored for optimization in real-valued parameter spaces. Therefore one should expect that they generally outperform the sGA on the sphere model. However, as we will see, when noise comes into play both algorithm classes exhibit similar behavior. But first, short definitions of the ES-algorithms used will be presented.

The $(1, \lambda)$-, the (μ, λ)-, and the $(\mu/\mu, \lambda)$-ES [21,22] using self-adaptation to tune the single component standard deviation σ of the mutation operator are used in the test. The mutations are produced by isotropic Gaussian random vectors $\mathbf{z}$

$$\mathbf{z} = (z_1, \ldots, z_N) := \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{E}) =: \vec{\mathcal{N}}(0,1). \tag{6}$$

Selection is performed by truncation which selects the μ best individuals out of λ offspring as parents for the next generation. The ES-algorithms can be expressed in a very compact form by the order notation

$$(\cdot)_{m;\lambda} \;:\quad \text{“select the } m\text{th best out of } \lambda\text{”} \tag{7}$$

which describes the selection used. Here, best refers to the objective of the optimization. It can stand for maximization as well as minimization; and

$$m;\lambda \;\in\; [1, \lambda]$$

is just the index number of the mth best individual.

Unlike sGAs, an individual genome in self-adaptive ESs does not only comprise the object parameter set $\mathbf{y}$ to be optimized, but also a strategy parameter set $\mathbf{s}$

$$\text{“individual”}_l := (\mathbf{y}_l, \mathbf{s}_l, \tilde{F}(\mathbf{y}_l)), \qquad l = 1 \ldots \lambda, \tag{8}$$

which is inherited together with the $\mathbf{y}$ vector. In case of isotropic Gaussian mutations, $\mathbf{s}$ is just the scalar σ used for the generation of the lth individual. Using these notations the (μ, λ)-ES with SA (self-adaptation) reads

$$\forall\, l = 1 \ldots \lambda \ : \begin{cases} r = \text{Random}[1 \ldots \mu] \\ \sigma_l^{(g+1)} := \sigma_{r;\lambda}^{(g)} \cdot \exp[\tau\, \mathcal{N}(0,1)] \\ \mathbf{y}_l^{(g+1)} := \mathbf{y}_{r;\lambda}^{(g)} + \sigma_l^{(g+1)} \vec{\mathcal{N}}(0,1). \end{cases} \tag{9}$$

Here, g is the generation counter and $r \in [1, \mu]$ is a random integer number sampled anew for each l. Furthermore, we have applied Schwefel's version of SA using log-normal mutations [23]. A reasonable choice for the learning parameter τ is

$$\tau \gtrapprox 1/\sqrt{N}. \tag{10}$$

Actually, for $(1, \lambda)$-ESs on the sphere one can prove [24] $\tau \sim c_{1,\lambda}/\sqrt{N}$, with the progress coefficient $c_{1,\lambda}$ (for its definition, see Eq. (58)).

The (μ, λ)-ES is a simple, but population based mutation-selection algorithm *without* recombination. In contrast to (μ, λ) the $(\mu/\mu, \lambda)$-ES uses recombination in its extreme form: the so-called multi-parent recombination [21,25]. There are two versions of μ/μ recombination. The *intermediate* one takes simply the average of all μ individuals to produce a new descendant. This version is recommended [26] for strategy parameter recombination. The second version is the *dominant* recombination [27], also known as global discrete recombination. It transfers randomly chosen parental y_i-components to the offspring resembling a generalized version of uniform crossover [20]. Dominant recombination is recommended for the treatment of object parameters when SA (self-adaptation) is desired. The self-adaptive $(\mu/\mu, \lambda)$-ES with isotropic Gaussian mutations reads

$$\forall\, l = 1 \ldots \lambda \ : \begin{cases} \sigma_l^{(g+1)} := \left(\frac{1}{\mu} \sum_{m=1}^{\mu} \sigma_{m;\lambda}^{(g)} \right) \cdot \exp[\tau\, \mathcal{N}(0,1)] \\ \\ \forall\, i = 1 \ldots N : \begin{cases} r_i = \text{Random}[1 \ldots \mu] \\ \left[\mathbf{y}_l^{(g+1)}\right]_i := \left[\mathbf{y}_{r_i;\lambda}^{(g)}\right]_i + \sigma_l^{(g+1)} \vec{\mathcal{N}}(0,1), \end{cases} \end{cases} \tag{11}$$

where $[\mathbf{y}]_i$ stands for the ith component of the vector $\mathbf{y}$; the normally distributed random generators $\mathcal{N}(0,1)$ are sampled anew for each i and l.

In order to make performance comparisons a suitable initialization of the ESs is to be chosen. To have a fair starting condition, the $\mathbf{y}_m^{(0)}$ vectors are concentrated in a randomly chosen point with a distance to the optimum which is roughly the $\langle R\rangle^{(0)}$ observed (i.e. measured) in the GA simulations of Figure 1: $\langle R\rangle^{(0)} \approx 0.57$. The choice of the initial $\sigma_m^{(0)} = 1.0$, however, is a "misplaced" one (for optimal local performance it should be around 0.1 in the example considered here); it was chosen with the intention to demonstrate the SA capabilities. Figure 2 shows the simulation results for strategies with $\lambda = 60$ (left pictures) and $\lambda = 120$ (right pictures).

The upper two curves in Figure 2 are obtained using the $(1,\lambda)$-ES. For the no noise case, $\sigma_\varepsilon = 0$, the ES approaches the optimum very fast and reduces $\langle R\rangle \to 0$, beating the sGA which gets stuck at a $R_\infty > 0$. However, even for the relatively small noise level $\sigma_\varepsilon = 0.1$ the sGA performs better. It is striking that $\langle F\rangle$ goes far above the noise free maximum at 1. Since only the "best", i.e. the largest fitness value produced by (4), contributes to $\langle F\rangle$, $\langle F\rangle = F_{1;\lambda}$, the observed behavior reflects just the outliers produced by the $\mathcal{N}(0,\sigma_\varepsilon^2)$ noise term. In other words, introducing *elitism* (i.e., keeping track of the best individual found so far) into noisy EAs will not qualitatively improve the convergence behavior. Elitism can even not exclude divergence. This remarkable property has been proven by Beyer [2] for the $(1+1)$-ES on the sphere model. From that paper we will also take the R_∞ formula which describes the steady state behavior of the noisy $(1,\lambda)$-ES (it appears also as a special case of a formula to be derived below, Eq. (89))

$$(1,\lambda)\text{-ES:} \quad R_\infty \geq \frac{1}{2}\sqrt{\frac{\sigma_\varepsilon N}{c_{1,\lambda}}}. \tag{12}$$

The equal sign holds for vanishing mutation strength $\sigma \to 0$. The $c_{1,\lambda}$ is the so-called progress coefficient (see Eq. (58), below) which basically is the expectation of the λth order statistics of the standard normal variate $\mathcal{N}(0,1)$.[2] We have $c_{1,60} \approx 2.32$ and $c_{1,120} \approx 2.57$. Although formula (12) exactly holds for the asymptotic case $N \to \infty$, it yields usable results even for $N = 10$ as one can verify in the upper two pictures of Figure 2.

Formula (12) explains most of the observations made so far. First, it shows that R_∞ monotonously increases with σ_ε. Second, R_∞ decreases with increasing population size λ, since $c_{1,\lambda} \sim \sqrt{2\ln\lambda}$ [2, p. 171]. That is, similar to the findings of Rattray and Shapiro [17] for GAs, the influence of the noise in $(1,\lambda)$-

[2] For an introduction into order statistics, the reader is referred to [28].

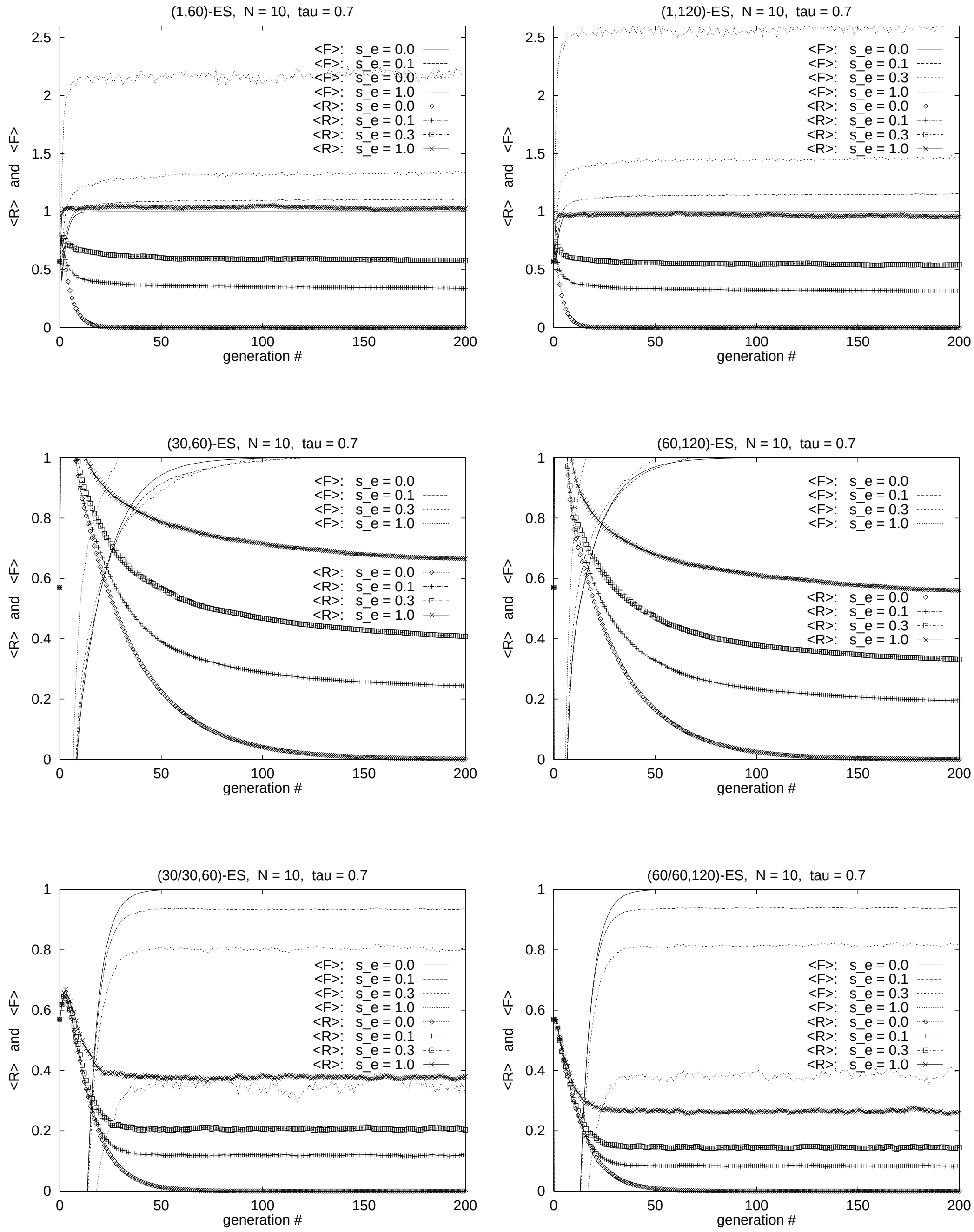

Fig. 2. ES dynamics of the $\langle F \rangle$ and $\langle R \rangle$ averages using the same noise levels as in Figure 1. Note, the upper two pictures use a larger vertical scale in order to display the $\langle F \rangle$ values. All populations start at $\langle R \rangle^{(0)} \approx 0.57$ indicated by the black box symbol. As to the $(30, 60)$-ES, the initial mutation strength $\sigma = 1.0$ is too large. One can see in the middle pictures that the population is initially driven away from the optimum (the $\langle R \rangle$ curves are coming from above). Note, the $\langle F \rangle$ curves in the two bottom pictures are fairly below the optimum. This is due to the high mutation strength σ generated by the SA.

ESs can be "removed" to a certain extend by choosing $\lambda \sim \exp\left[\left(\frac{N}{\sqrt{2}\, 4R_\infty}\right)^2 \sigma_\varepsilon^2\right]$.

Third, the residual distance increases with the square root of the parameter space dimension. This is bad news, because it indicates that, given a fixed noise level, higher dimensional problems are harder to be optimized.

Although not explicitly tested, it is reasonable to assume that GAs suffer from the same problem. Considering the GA population sizing model of Goldberg et al. [12] one comes to the same qualitative statement. In their population sizing equation

$$\lambda = 2c\kappa \frac{\sigma_M^2 + \sigma_\varepsilon^2}{d^2} \tag{13}$$

σ_M^2 is the variance of the collateral and σ_ε^2 of the fitness noise, d measures the "signal difference" between competing schemata, and c is a parameter depending on the significance level of the building block decision. What is of interest here, apart from the σ_ε^2 influence which is also consonant with the observations in our ES experiments, is the κ which counts the number of competing schemata. Assuming that κ should be related to the parameter space dimension, e.g. $\kappa \geq \mathrm{const.}N$, it becomes clear that the population size λ must be scaled with N. (There is a more refined sizing model by Harik et al. [13] to be compared with results from recombinative ESs, see the discussion after Eq. (18).)

Using only the "best" individual out of λ is obviously not the most clever policy. Switching to parent numbers $\mu > 1$ can considerably improve the convergence behavior, as can be inferred from the middle pictures of Figure 2. Furthermore, using μ/μ recombination gives an additional performance gain. Up to now there is no theory for calculating R_∞ for those cases. However, some of the general tendencies from (12) can be easily verified by experiments: one finds $R_\infty \propto \sqrt{\sigma_\varepsilon N}$. Putting all our knowledge and intuition together, the R_∞ formulae for the general sphere model (5) might read

$$(\mu, \lambda)\text{-ES:} \quad R_\infty \overset{?}{\geq} \sqrt[\alpha]{\frac{\sigma_\varepsilon N}{2c\alpha\sqrt{\mu}c_{\mu,\lambda}}} \tag{14}$$

and

$$(\mu/\mu, \lambda)\text{-ES:} \quad R_\infty \overset{?}{\geq} \sqrt[\alpha]{\frac{\sigma_\varepsilon N}{2c\alpha\mu c_{\mu/\mu,\lambda}}}, \tag{15}$$

where $c_{\mu,\lambda}$ and $c_{\mu/\mu,\lambda}$ are progress coefficients defined in [25] and [27], respectively. The predictive power of (14) is relatively high, even for $N = 10$. Equation (15) works satisfactorily, though not as well as (14); however, it correctly describes the μ scaling behavior (at least for $\alpha = 2$).

Apart from the question of the steady state value R_∞, the speed of convergence is of interest. It is defined as the expected change from generation g to $g+1$ and is called *progress rate* φ

$$\varphi := \mathrm{E}\left[\langle R\rangle^{(g)} - \langle R\rangle^{(g+1)}\right]. \tag{16}$$

Large φ values produce steeper falling $\langle R\rangle^{(g)}$ curves and thus, they are desirable. As to the progress rate, (μ, λ)-ESs are slower than $(\mu/\mu, \lambda)$-ESs, provided that the σSA works correctly. Calculating φ for a given fitness model is one of the main goals of ES theory. As an example, the $(1, \lambda)$-ES with rescaled mutations will be treated in Section 5.

4 Guidelines for improving the EA performance under noise

One might argue that driving guidelines for general EA performance from the sphere model is somewhat bold. However, there are at least three arguments that should be taken into account.

First, as we have seen, noise deteriorates the final location of the optimum ($R_\infty > 0$). That is, it mainly affects the end phase of the optimization. In this phase, the EA has usually "decided" for an attractor in the multimodal fitness landscape. Considering the $\mathbb{R}^N$, such attractors can be locally approximated by a bilinear form which itself can often be approximated by a sphere model with a certain mean curvature [27, pp. 102 – 108].

Second, the analytical formulae obtained from the sphere model allow for a discussion of the parameters influencing the behavior of the EA. There is a certain "hope" (to be verified by experiments) that some of the properties derived can be "saved" and transferred to more complicated fitness functions and real-world applications.

Third, we know by theory how the EA *should* perform on our test function. If there is a large discrepancy between the theoretical predictions and the performance of the real algorithm, then one *can* suspect an error in the theory or (and) a wrongly working algorithm. Indeed, this is the motor of further developments.

Furthermore, we will see that some of the performance improving techniques can also be inferred from empirical GA research and GA theory based on building block competition models.

There are different measures to improve the performance of noisy EAs:

(1) resampling,
(2) sizing of the population size λ (and μ),
(3) inheritance of rescaled mutations.

They will be discussed in the next two subsections.

4.1 Resampling and the sizing of the population

Resampling is a simple measure to improve the convergence (toward the optimum) of EAs. Given the individual's genome $\mathbf{y}_l$, the fitness $F(\mathbf{y}_l)$ is measured m times and averaged yielding a fitness

$$\overline{F}(\mathbf{y}_l) = \frac{1}{m}\sum_{k=1}^{m} F(\mathbf{y}_l), \quad \mathbf{y}_l = \text{const.} \quad \Rightarrow \quad \overline{\sigma}_\varepsilon = \sqrt{\mathrm{Var}[\overline{F}(\mathbf{y}_l)]} = \frac{\sigma_\varepsilon}{\sqrt{m}}. \tag{17}$$

That is, the noise strength of $\overline{F}$ is reduced by a factor $\sqrt{m}$.

Assuming that the fitness calculation is the most time consuming part of the EA, the resampling technique does not come for free. Alternatively, one might also raise the population size λ by the factor m. Fitzpatrick's and Grefenstette's findings [3] point in this direction. However, as to the $(1,\lambda)$-ES the opposite does usually hold. The $(1,\lambda)$-ES with m-times resampling gives better results than the $(1, m\cdot\lambda)$-ES, except for the cases $\lambda = 2$, $m < 12$ and $\lambda = 3$, $m < 4$ [2].

Things will change when a parental population with $\mu > 1$ is used. The effects obtained by choosing $\mu > 1$ are almost for free (as to the fitness calculations). Provided that (14) is correct, then the effect of the population in (μ,λ)-ESs compared to the generalized form of (12), $R_\infty \geq \sqrt[\alpha]{\sigma_\varepsilon N/2c\alpha c_{1,\lambda}}$, is of the same order as for a $(1,\lambda)$-ES with m-times resampling. This is so, because one can interpret $\sigma_\varepsilon/\sqrt{\mu}$ in (14) as a "μ-times resampling" implicitly performed by the parental population of size μ. Using the generalized form of (12) and (14) we have

$$\begin{aligned}
(1,\lambda)\text{-ES with } \mu\text{-times resampling:} \quad & R_\infty \geq \sqrt[\alpha]{\frac{\sigma_\varepsilon N}{2c\alpha\sqrt{\mu}c_{1,\lambda}}} \\
(\mu, \mu\cdot\lambda)\text{-ES:} \quad & R_\infty \geq \sqrt[\alpha]{\frac{\sigma_\varepsilon N}{2c\alpha\sqrt{\mu}c_{\mu,\mu\lambda}}}.
\end{aligned}$$

Since $c_{1,\lambda} \leq c_{\mu,\mu\lambda}$ can be assumed [25], the $(\mu, \mu\cdot\lambda)$-ES should perform slightly better than the $(1, \lambda)$-ES with resampling.

There remains the question how to choose μ given fixed population size λ. Since our main goal is to minimize R_∞, applying (14) will lead us to the condition $\sqrt{\mu}c_{\mu,\lambda} \to \max_\mu$. The function $f(\mu) = \sqrt{\mu}c_{\mu,\lambda}$ has a relatively broad maximum such that the choice of μ is uncritical as long as μ and $\lambda - \mu$ are sufficiently large. In the asymptotic limit, $\lambda \to \infty$, $\mu \to \infty$ there is the conjecture that $c_{\mu,\lambda} \sim \sqrt{2 \ln \lambda/\mu}$. Thus, one can calculate the optimum $\hat{\mu}$ by maximizing $\mu \ln \lambda/\mu$, with the result $\hat{\mu} \sim \lambda/\mathrm{e}$. Experiments are still to be performed in order to test the applicability of this formula.

Recombination seems to bring an additional convergence gain, as can be seen from formula (15) compared to (14). This additional gain is mainly a result of similarity extraction (the so-called *genetic repair effect* [29]) which takes place in the parameter space. One can again ask for the optimum $\hat{\mu}$. Since there is an asymptotically *exact* $c_{\mu/\mu,\lambda}$ formula [27, p. 92]

$$c_{\mu/\mu,\lambda} \simeq \frac{\lambda}{\mu} \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\Phi^{-1}\left(1 - \frac{\mu}{\lambda}\right)\right)^2\right] \tag{18}$$

and the minimal R_∞ in (15) is reached for $\mu c_{\mu/\mu,\lambda} \to \max_\mu$, we find the condition $\Phi^{-1}(1 - \hat{\mu}/\lambda) = 0$. Here, Φ^{-1} is the inverse function to the cdf (cumulative distribution function) of the standard normal variate $\mathcal{N}(0, 1)$. Since $\Phi(0) = 1/2$, one finds $\frac{1}{2} = \frac{\hat{\mu}}{\lambda}$ and therefore $\hat{\mu} \simeq \lambda/2$. Note, this $\hat{\mu}$ considerably deviates from the recommended $\hat{\mu} \approx \lambda/7$ [30] for the noiseless case.[3] Hammel and Bäck [4, p. 165] used a strategy with $\mu = 15$ and $\lambda = 100$ leading to $\hat{\mu} \approx \lambda/6.67$ and concluded that resampling should be preferred. Their conclusions might be the result of a wrong population sizing. Further simulations should be carried out to get more definite answers as to the question resampling vs. population up-sizing.

Choosing the right population size can also be discussed from the viewpoint of building block decision-making in GAs. This approach by Goldberg et al. has already been mentioned in Point 3.2.2. A more refined model has been proposed by Harik et al. (1997) [13]. The sizing equation reads

$$\lambda = -2^{k-1} \ln(p) \frac{\sigma_{BB}\sqrt{2m'}}{d}, \tag{19}$$

where k is the order of the building block, p is something like a failure probability for selecting a wrong building block, m' is one less than the number of build-

[3] To be exact, there is *no* fixed optimal $\lambda/\hat{\mu}$ ratio; $\hat{\mu}$ depends on N and λ. However, for $N \to \infty$, $\lambda \to \infty$ one asymptotically finds $\hat{\mu} \approx \lambda/3.7$ on the sphere model [27].

ing blocks in the individual's binary genome (bitstring). For our discussion, the parameters of interest are σ_{BB} and d, where σ_{BB}^2 is the average building block variance and d is the so-called signal difference. The σ_{BB} can comprise different noise sources, thus, it can also contain the fitness noise σ_ε. The scaling property of this model with respect to σ_{BB} and d reads $\lambda = \mathcal{O}(\sigma_{BB}/d)$. It is interesting to see that the same scaling property is obtained from Eq. (15) in case of an optimal $(\mu/\mu, \lambda)$-ES: With $\mu = \lambda/2$, $Q_\infty := cR_\infty^\alpha$, and using results mentioned above one finds $\lambda \simeq \sqrt{\pi}\sigma_\varepsilon N/\alpha Q_\infty \sqrt{2}$. If one interprets Q_∞, i.e. the expected fitness distance to the optimal fitness, as signal difference d then a similar scaling behavior, $\lambda = \mathcal{O}(\sigma_\varepsilon/Q_\infty)$, is obtained.[4] It is an open question whether this functional similarity in the scaling properties of the population size in GAs and recombinative ESs is just an incidence or reveals a deeper connection.

4.2 Inheritance of rescaled mutations

Unlike resampling, the technique to be presented here does not require additional fitness evaluations. However, it is restricted to (quasi-) continuous search spaces, such as the $\mathbb{R}^N$, because it requires a down-scaling of the mutations. It has been proposed by Ostermeier and Rechenberg [22, p. 195]. An asymptotic analysis for the $(1, \lambda)$ version can be found in [31], its limitations as well as first implementation issues are published in [32]. In this paper, the N-dependent progress rate analysis will be performed in Section 5 and an improved SA method will be proposed in Section 6.

The idea for the $(1, \lambda)$ version is to perform large mutations $\mathbf{z}_l$ $(l = 1 \dots \lambda)$ from the parental state $\mathbf{y}_\mathrm{p}$. Due to the large mutations, most of the offspring will have worse fitness than their parent. The mutation $\mathbf{z}_{1;\lambda}$ which produced the best offspring, however, is rescaled in length by a factor κ

$$\mathbf{y}_\mathrm{p}^{(g+1)} := \mathbf{y}_\mathrm{p}^{(g)} + \frac{1}{\kappa}\mathbf{z}_{1;\lambda}^{(g)}, \qquad \kappa > 1. \tag{20}$$

Thus, the $(1, \lambda)$-ES can perform large search steps with the result of larger fitness differences which will (hopefully) be significant over the noise level. Having found the right direction, the ES makes only a reduced size step (20) in that direction. The result serves as parent for the next generation.

In order to fit the algorithm into a frame similar to (9) we will formulate it in offspring notation. This can be easily done because the offspring are generated

[4] The author is grateful to D. E. Goldberg who pointed him to this functional similarity.

according to $\mathbf{y}_l^{(g)} := \mathbf{y}_\mathrm{p}^{(g)} + \mathbf{z}_l^{(g)}$ and $\mathbf{y}_l^{(g+1)} := \mathbf{y}_\mathrm{p}^{(g+1)} + \mathbf{z}_l^{(g+1)}$. With (20) one finds

$$\mathbf{y}_l^{(g+1)} = \mathbf{y}_\mathrm{p}^{(g+1)} + \mathbf{z}_l^{(g+1)} = \mathbf{y}_\mathrm{p}^{(g)} + \frac{1}{\kappa}\mathbf{z}_{1;\lambda}^{(g)} + \mathbf{z}_l^{(g+1)} \tag{21}$$

and from $\mathbf{y}_{1;\lambda}^{(g)} = \mathbf{y}_\mathrm{p}^{(g)} + \mathbf{z}_{1;\lambda}^{(g)}$ one obtains $\mathbf{y}_\mathrm{p}^{(g)} = \mathbf{y}_{1;\lambda}^{(g)} - \mathbf{z}_{1;\lambda}^{(g)}$. After substitution into (21) one gets

$$\mathbf{y}_l^{(g+1)} = \mathbf{y}_{1;\lambda}^{(g)} + \left(\frac{1}{\kappa} - 1\right)\mathbf{z}_{1;\lambda}^{(g)} + \mathbf{z}_l^{(g+1)}. \tag{22}$$

Thus, we have expressed the new offspring by the offspring and the mutations of the previous generation. Now we can formulate the self-adaptive $(1,\lambda)$-ES with rescaled mutations

$$\forall\, l = 1 \ldots \lambda \;:\; \begin{cases} \xi_l^{(g+1)} := \exp[\tau\, \mathcal{N}(0,1)] \\ \sigma_l^{(g+1)} := \sigma_{1;\lambda}^{(g)}\, \xi_l^{(g+1)} \\ \mathbf{z}_l^{(g+1)} := \sigma_l^{(g+1)} \vec{\mathcal{N}}(0,1) \\ \mathbf{y}_l^{(g+1)} := \mathbf{y}_{1;\lambda}^{(g)} + \left(\frac{1}{\kappa} - 1\right)\mathbf{z}_{1;\lambda}^{(g)} + \mathbf{z}_l^{(g+1)}. \end{cases} \tag{23}$$

Expressing the $(1,\lambda)$-ES in this way, the generalization to the (μ,λ)-ES is straightforward. However, the (μ,λ) version has not been intensively tested in experiments. Its investigation remains for future research.

In the next section it will be shown that the rescaling rule (20) of algorithm (23) allows for a $R_\infty \xrightarrow{\kappa\to\infty} 0$ as far as the sphere model is concerned. However, the *real* algorithm (23) exhibits only poor performance in ES experiments. That is, there must be lines in (23) which do not work as usually expected. The discussion of this interesting issue will be postponed to Section 6.

5 The progress rate analysis of the rescaled mutation technique

This section provides the calculation of the progress rate φ on the N-dimensional sphere model. It is a very technical section. Readers just interested in the basic ideas of the progress rate analysis should read Point 5.1.1 and then immediately skip to the result, Eq. (79). Where we will again establish a link to GA theory. The section is organized as follows. First, the general ideas of the approach are presented. After that, the calculation of $\overline{x}_{1;\lambda}$ and $\overline{\mathbf{h}^2}_{1;\lambda}$, to be defined below, will be performed and then, the parts will be put together. Having the approximative φ formula, its predictive power will be tested in some experiments (Section 5.2) showing that it yields far better results than the asymptotic formula derived in [31,32]. Finally, in Section 5.3 the convergence behavior of the $(1, \lambda)$-ES will be discussed.

5.1 On the derivation of φ

5.1.1 The general approach

The progress rate φ as the expected parental distance change has already been defined by Eq. (16). Since there is only one parent $\mathbf{y}_\mathrm{p}$ in $(1, \lambda)$-ESs, one has at generation g

$$\varphi_{1,\lambda} = \mathrm{E}\left\{\|\mathbf{y}_\mathrm{p}^{(g)} - \hat{\mathbf{y}}\| - \|\mathbf{y}_\mathrm{p}^{(g+1)} - \hat{\mathbf{y}}\|\right\} = \mathrm{E}\left\{\|\mathbf{R}\| - \|\tilde{\mathbf{R}}\|\right\}. \tag{24}$$

The new parental state at $g+1$ is given by (20). Let us decompose the mutation $\mathbf{z}_{1;\lambda}^{(g)}$ into a component x pointing in direction of the optimum $\hat{\mathbf{y}}$ and a residual vector $\mathbf{h}$ perpendicular to the x part

$$\mathbf{z}_{1;\lambda}^{(g)} := -x_{1;\lambda}\mathbf{e}_R + \mathbf{h}_{1;\lambda}, \qquad \text{with} \qquad \mathbf{e}_R^\mathrm{T}\mathbf{h}_{1;\lambda} = 0. \tag{25}$$

This decomposition is depicted in the left picture of Figure 3 which already takes the rescaling into account. If (25) is inserted into (20), and the $\mathbf{y}_\mathrm{p}^{(g+1)}$ in

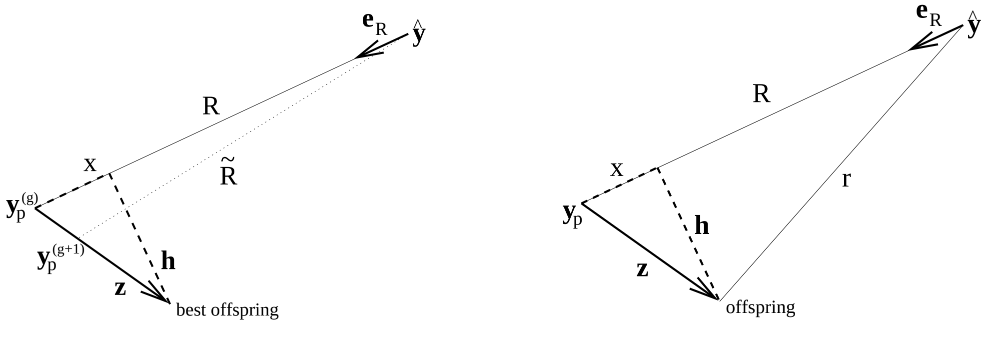

Fig. 3. Decomposition of the $\mathbf{z}$ mutations in a part x in optimum direction, defined by the unity vector $\mathbf{e}_R$ and a perpendicular $\mathbf{h}$ vector. The left picture displays the rescaling of the best mutation, whereas the right one depicts the decomposition of an arbitrary mutation.

(24) is substituted by (20), one obtains (omitting the generation counter from now on)

$$\varphi = \mathrm{E}\left\{\|\mathbf{R}\| - \left\|\mathbf{R} + \frac{1}{\kappa}\mathbf{z}_{1;\lambda}\right\|\right\} = E\left\{\|\mathbf{R}\| - \left\|R\mathbf{e}_R - \frac{1}{\kappa}x_{1;\lambda}\mathbf{e}_R + \frac{1}{\kappa}\mathbf{h}_{1;\lambda}\right\|\right\}. \quad (26)$$

Recalling that $R = \|\mathbf{R}\|$ one gets

$$\varphi_{1,\lambda} = \mathrm{E}\left\{R - \sqrt{\left(R - \frac{1}{\kappa}x_{1;\lambda}\right)^2 + \frac{1}{\kappa^2}\mathbf{h}^2_{1;\lambda}}\right\}. \quad (27)$$

Up to this point, the $\varphi_{1,\lambda}$ formula does hold for all fitness functions. Even the next step, introducing normalized quantities φ^* and σ^*

$$\varphi^* := \varphi\frac{N}{R} \qquad \text{and} \qquad \sigma^* := \sigma\frac{N}{R}, \quad (28)$$

is independent of the fitness model used; however, it is meaningful and only intended for the N-dimensional sphere model. With (27), the normalized progress rate φ^* reads

$$\varphi^*_{1,\lambda} = N\,\mathrm{E}\left\{1 - \sqrt{\left(1 - \frac{1}{\kappa N}\frac{N}{R}x_{1;\lambda}\right)^2 + \frac{1}{\kappa^2}\frac{\mathbf{h}^2_{1;\lambda}}{R^2}}\right\}. \quad (29)$$

This expectation expression depends on two scalar random variates $x_{1;\lambda}$ and $\mathbf{h}^2_{1;\lambda}$. Their pdfs (probability density functions) depend on the fitness model, the mutation distribution, the parameter space dimension N, the population size λ, and the actual parental state $\mathbf{y}_{\mathrm{p}}^{(g)}$. There is no hope to get a closed analytical expression for (29) and $N < \infty$. Therefore, approximations must be developed.

The first approximation to be introduced removes the outer $\mathrm{E}\{\}$

$$\varphi^*_{1,\lambda} = N\left[1 - \sqrt{\left(1 - \frac{1}{\kappa N}\frac{N}{R}\overline{x}_{1;\lambda}\right)^2 + \frac{1}{\kappa^2}\frac{\overline{\mathbf{h}^2}^2_{1;\lambda}}{R^2}}\right] + \dots \quad (30)$$

by neglecting the fluctuations around the mean values

$$\overline{x}_{1;\lambda} := \mathrm{E}\{x_{1;\lambda}\} \qquad \text{and} \qquad \overline{\mathbf{h}^2}_{1;\lambda} := \mathrm{E}\{\mathbf{h}^2_{1;\lambda}\}. \quad (31)$$

This is asymptotically correct for Gaussian mutations with $\lambda < \infty$ and $N \to \infty$: $\mathbf{h}^2$ is the sum of $N-1$ squared and independent $\mathcal{N}(0, \sigma^2)$ random variates. Therefore, the central limit theorem holds and $\sqrt{\mathrm{Var}\{\mathbf{h}^2\}}/\mathrm{E}\{\mathbf{h}^2\} \to 0$ is fulfilled for each (single) mutation $\mathbf{z}$. Similar arguments do hold for $\mathrm{Var}\{x\} = \sigma^2$

which is small compared to $\overline{\mathbf{h}^2} \sim (N-1)\sigma^2$ and R. The exact proof that requires the Taylor expansion of (29) at the values of (31) will be omitted here.

With (30), we are left with the problem of determining $\overline{x}_{1;\lambda}$ and $\overline{\mathbf{h}^2}_{1;\lambda}$. An intermediate step, however, should be taken before: the calculation of the pdf of the mutation-induced noisy fitness distribution.

5.1.2 On the calculation of the mutation-induced noisy fitness distribution $p(\tilde{Q})$

At this point the fitness model enters the stage. We will analyze the minimization task for model (5). In order to keep the calculations as simple as possible, the case $\alpha = 2$ is considered.

Let $Q(r) := cr^\alpha$ be the noise-free fitness of an offspring generated from a parental state $\mathbf{y}_\mathrm{p}$ by the mutation $\mathbf{z}$ (see right picture of Figure 3). According to (3) and (5) the conditional pdf of the offspring's noise perturbed fitness reads

$$p(\tilde{Q}|Q(r)) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \exp\left[-\frac{1}{2}\left(\frac{\tilde{Q}-Q(r)}{\sigma_\varepsilon}\right)^2\right]. \tag{32}$$

If the pdf of the offspring's distance-to-optimum r were known one could calculate the pdf of $\tilde{Q}$

$$p(\tilde{Q}) = \int p(\tilde{Q}|Q(r))p(r)\, dr. \tag{33}$$

The general calculation of $p(r)$ is difficult, it can be found in [25]. However, due to the restriction to $\alpha = 2$, it suffices to calculate $p(r^2)$. To this end let us consider the right picture of Figure 3. By geometry one reads

$$r^2 = (R-x)^2 + \mathbf{h}^2 = R^2 - 2Rx + x^2 + \mathbf{h}^2. \tag{34}$$

Due to the spherical symmetry of the mutations $\mathbf{z}$ (not to be confused with the sphere model assumption), the components of the new decomposition in Figure 3 obey the same normal distribution density as the original $\mathbf{z}$-components, i.e. $x = \mathcal{N}(0,\sigma^2)$ and $(\mathbf{h})_i = \mathcal{N}(0,\sigma^2)$. Thus, r^2 can be interpreted as a sum of a χ^2-distribution and a normal distribution $\mathcal{N}(R^2, 4R^2\sigma^2)$. Since the χ^2-distribution can be well approximated by a normal distribution, r^2 itself will be approximately normal. The simplest way to get the mean and standard deviation of the normal approximation is to directly calculate $\overline{r^2}$ and

$\sigma_{r^2} = \sqrt{\overline{r^4} - (\overline{r^2})^2}$. Knowing that the first four moments of the random variate $x = \mathcal{N}(0, \sigma^2)$ are $\overline{x} = 0$, $\overline{x^2} = \sigma^2$, $\overline{x^3} = 0$, and $\overline{x^4} = 3\sigma^4$ one obtains by simple calculations and with (28)

$$\overline{r^2} = R^2 + N\sigma^2 \qquad \text{and} \qquad \sigma_{r^2} = \sqrt{4R^2\sigma^2 + 2N\sigma^4} = 2\sigma R\sqrt{1 + \frac{\sigma^{*2}}{2N}}. \tag{35}$$

Hence, the normal approximation of $p(r^2)$ reads

$$p(r^2) = \frac{1}{\sqrt{2\pi}\sigma_{r^2}} \exp\left[-\frac{1}{2}\left(\frac{r^2 - (R^2 + N\sigma^2)}{\sigma_{r^2}}\right)^2\right]. \tag{36}$$

Now we can calculate $p(\tilde{Q})$ by inserting (36) and (32) into (33) using $s := r^2$ instead of r and taking (5) into account

$$p(\tilde{Q}) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \frac{1}{\sqrt{2\pi}\sigma_{r^2}} \int \exp\left[-\frac{1}{2}\left(\frac{\tilde{Q} - cs^{\alpha/2}}{\sigma_\varepsilon}\right)^2\right] \exp\left[-\frac{1}{2}\left(\frac{s - (R^2 + N\sigma^2)}{\sigma_{r^2}}\right)^2\right] ds. \tag{37}$$

A closed integration is possible for $\alpha = 2$ only. Other α values must be treated by series expansions (not presented here). In order to calculate (37) $t := [s - (R^2 + N\sigma^2)]/\sigma_{r^2}$ is substituted, leading to

$$p(\tilde{Q}) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}t^2} \exp\left[-\frac{1}{2}\left(-\frac{c\sigma_{r^2}}{\sigma_\varepsilon}t + \frac{\tilde{Q} - c(R^2 + N\sigma^2)}{\sigma_\varepsilon}\right)^2\right] dt \tag{38}$$

The integral can be solved because

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}t^2} \exp\left[-\frac{1}{2}(at + b)^2\right] dt = \frac{1}{\sqrt{1 + a^2}} \exp\left(-\frac{1}{2}\frac{b^2}{1 + a^2}\right). \tag{39}$$

Using $\tilde{\sigma}$ as symbol for the standard deviation of the noisy offspring fitness, we finally obtain

$$p(\tilde{Q}) = \frac{1}{\sqrt{2\pi}\tilde{\sigma}} \exp\left[-\frac{1}{2}\left(\frac{\tilde{Q} - c(R^2 + N\sigma^2)}{\tilde{\sigma}}\right)^2\right] \tag{40}$$

with

$$\tilde{\sigma} := \sqrt{\sigma_\varepsilon^2 + (2cR\sigma)^2 + 2c^2N\sigma^4}. \tag{41}$$

The cdf (cumulative distribution function) of $\tilde{Q}$ reads

$$P_1(\tilde{Q}) = \Phi\left[\frac{\tilde{Q} - c(R^2 + N\sigma^2)}{\tilde{\sigma}}\right]. \tag{42}$$

Note, $\Phi(x)$ is the cdf of the standard normal distribution $\mathcal{N}(0,1)$. It is connected to the error function by $\Phi(x) = \frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)$.

5.1.3 On the calculation of $\overline{x}_{1;\lambda}$

In order to calculate $\overline{x}_{1;\lambda}$ we need the pdf of $x_{1;\lambda}$ which will be denoted by $p_{1;\lambda}(x)$. Each of the λ mutants generated can be the best, provided that it is *accepted* as the best. This leads us to

$$p_{1;\lambda}(x) = \lambda p_x(x) P_{\mathrm{a}1,\lambda}(x). \tag{43}$$

Since the x component of the mutation $\mathbf{z}$ is $\mathcal{N}(0,\sigma^2)$ distributed, we have

$$p_x(x) = \frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2}. \tag{44}$$

This is due to the isotropy of the mutations used. λ counts the number of different possibilities of being the best. It remains the determination of the acceptance probability $P_{\mathrm{a}1,\lambda}(x)$. To this end, consider the probability for the fitness being in the interval $(\tilde{Q}_{|x} - d\tilde{Q},\ \tilde{Q}_{|x}]$ given a certain value of x. It is $p(\tilde{Q}_{|x}|x)d\tilde{Q}$. In order to be the best, the remaining $\lambda - 1$ mutants must have $\tilde{Q}$ values which are larger than $\tilde{Q}_{|x}$ (minimization is considered). This occurs for a single individual with the probability $\mathrm{P}(\tilde{Q} > \tilde{Q}_{|x}) = 1 - \mathrm{P}(\tilde{Q} \le \tilde{Q}_{|x}) = 1 - P_1(\tilde{Q}_{|x})$, where P_1 is given by (42). Since there are $\lambda - 1$ independent mutants which must fulfill $\tilde{Q} > \tilde{Q}_{|x}$, one gets the probability $p(\tilde{Q}_{|x}|x)d\tilde{Q}\,[1 - P_1(\tilde{Q}_{|x})]^{\lambda-1}$ and hence (writing $\tilde{Q}$ instead of $\tilde{Q}_{|x}$)

$$P_{\mathrm{a}1,\lambda}(x) = \int_{-\infty}^{\infty} p(\tilde{Q}|x)[1 - P_1(\tilde{Q})]^{\lambda-1}\,d\tilde{Q}. \tag{45}$$

The conditional pdf $p(\tilde{Q}|x)$ is to be determined next. Recalling (32) and (34), the noise distribution is a conditional pdf w.r.t. x *and* $\mathbf{h}$. Neglecting the x^2

term in (34) (in order to get tractable integrals), which is allowed because $\mathrm{E}\{x^2\}/\mathrm{E}\{\mathbf{h}^2\} \sim 1/N$, one obtains

$$p(\tilde{Q}|x, \mathbf{h}^2) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \exp\left[-\frac{1}{2}\left(\frac{\tilde{Q} - cR^2 + 2cRx - c\,\mathbf{h}^2}{\sigma_\varepsilon}\right)^2\right]. \tag{46}$$

That is, given the pdf $p_\mathbf{h}(\mathbf{h}^2)$ one can calculate $p(\tilde{Q}|x)$

$$p(\tilde{Q}|x) = \int p(\tilde{Q}|x, \mathbf{h}^2)\, p_\mathbf{h}(\mathbf{h}^2)\, d(\mathbf{h}^2). \tag{47}$$

The $p_\mathbf{h}(\mathbf{h}^2)$ is a χ^2-distribution with $N-1$ degrees of freedom. Again, we apply the normal approximation method which had led to (36). One easily finds $\overline{\mathbf{h}^2} = (N-1)\sigma^2$ and $\mathrm{Var}\{\mathbf{h}^2\} = 2(N-1)\sigma^4$. To get tractable integrals, we are forced to use N instead of $N-1$ in the sequel. The error made by this approximation is of order $1/N$ and can thus be neglected for sufficiently large N. The pdf reads

$$p_\mathbf{h}(\mathbf{h}^2) = \frac{1}{\sqrt{2\pi}\sqrt{2N}\sigma^2} \exp\left[-\frac{1}{2}\left(\frac{\mathbf{h}^2 - N\sigma^2}{\sqrt{2N}\sigma^2}\right)^2\right]. \tag{48}$$

Now, the integration (47) can be performed. Using (46) and (48) and writing $u := \mathbf{h}^2$ we obtain

$$\begin{aligned} p(\tilde{Q}|x) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \frac{1}{\sqrt{2\pi}\sqrt{2N}\sigma^2} \int \exp\left[-\frac{1}{2}\left(\frac{\tilde{Q} - cR^2 + 2cRx - c\,\mathbf{h}^2}{\sigma_\varepsilon}\right)^2\right] \\ \times \exp\left[-\frac{1}{2}\left(\frac{u - N\sigma^2}{\sqrt{2N}\sigma^2}\right)^2\right] du. \end{aligned} \tag{49}$$

After the substitution $t := (u - N\sigma^2)/\sqrt{2N}\sigma^2$ the integral reads

$$p(\tilde{Q}|x) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \frac{1}{\sqrt{2\pi}} \int\limits_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}t^2} \exp\left[-\frac{1}{2}\left(-\frac{c\sqrt{2N}\sigma^2}{\sigma_\varepsilon}t + \frac{\tilde{Q} - cR^2 + 2cRx - cN\sigma^2}{\sigma_\varepsilon}\right)^2\right] dt. \tag{50}$$

Applying (39) we get

$$p(\tilde{Q}|x) = \frac{1}{\sqrt{2\pi}\tilde{\sigma}_x} \exp\left[-\frac{1}{2}\left(\frac{\tilde{Q} - cR^2 + 2cRx - cN\sigma^2}{\tilde{\sigma}_x}\right)^2\right] \tag{51}$$

with

$$\tilde{\sigma}_x := \sqrt{\sigma_\varepsilon^2 + 2c^2 N\sigma^4}. \tag{52}$$

This result is to be inserted into (45). Substituting $t := -[\tilde{Q} - c(R^2 + N\sigma^2)]/\tilde{\sigma}$, considering (42), and taking $\Phi(t) = 1 - \Phi(-t)$ into account, one obtains

$$P_{\mathrm{a}1,\lambda}(x) = \frac{\tilde{\sigma}}{\tilde{\sigma}_x}\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}[\Phi(t)]^{\lambda-1}\exp\left[-\frac{1}{2}\left(-\frac{\tilde{\sigma}}{\tilde{\sigma}_x}t + \frac{2cRx}{\tilde{\sigma}_x}\right)^2\right]dt. \tag{53}$$

The expected value of $x_{1;\lambda}$ is obtained from (43) with (44) and (53)

$$\overline{x}_{1;\lambda} = \frac{\tilde{\sigma}}{\tilde{\sigma}_x}\frac{\lambda}{2\pi\sigma}\int_{-\infty}^{\infty} x\mathrm{e}^{-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2}\int_{-\infty}^{\infty}[\Phi(t)]^{\lambda-1}\exp\left[-\frac{1}{2}\left(\frac{2cR}{\tilde{\sigma}_x}x - \frac{\tilde{\sigma}}{\tilde{\sigma}_x}t\right)^2\right]dt\,dx. \tag{54}$$

After changing the order of integration and substituting $s := x/\sigma$ one gets

$$\overline{x}_{1;\lambda} = \frac{\tilde{\sigma}}{\tilde{\sigma}_x}\frac{\sigma\lambda}{\sqrt{2\pi}}\int_{-\infty}^{\infty}[\Phi(t)]^{\lambda-1}\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} s\mathrm{e}^{-\frac{1}{2}s^2}\exp\left[-\frac{1}{2}\left(\frac{2cR\sigma}{\tilde{\sigma}_x}s - \frac{\tilde{\sigma}}{\tilde{\sigma}_x}t\right)^2\right]ds\,dt \tag{55}$$

The inner integration can be carried out; as can be shown

$$\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} s\mathrm{e}^{-\frac{1}{2}s^2}\exp\left[-\frac{1}{2}(as+b)^2\right]ds = \frac{-ab\exp\left(-\frac{1}{2}\frac{b^2}{1+a^2}\right)}{\sqrt{1+a^2}(1+a^2)} \tag{56}$$

does hold. After a simple calculation, taking the definition of $\tilde{\sigma}$ (41) and of $\tilde{\sigma}_x$ (52) into account, one obtains

$$\overline{x}_{1;\lambda} = \frac{2cR\sigma^2}{\tilde{\sigma}}\frac{\lambda}{\sqrt{2\pi}}\int_{-\infty}^{\infty} t\mathrm{e}^{-\frac{1}{2}t^2}[\Phi(t)]^{\lambda-1}dt, \tag{57}$$

that contains the well-known *progress coefficient*

$$c_{1,\lambda} := \frac{\lambda}{\sqrt{2\pi}}\int_{-\infty}^{\infty} t\mathrm{e}^{-\frac{1}{2}t^2}[\Phi(t)]^{\lambda-1}dt. \tag{58}$$

After introduction of the *normalized noise strength* σ_ε^* [2]

$$\sigma_\varepsilon^* := \frac{\sigma_\varepsilon}{|Q'(R)|}\frac{N}{R} \qquad \text{with} \qquad Q'(R) := \frac{dQ}{dR} \tag{59}$$

which specializes with the noise free fitness $Q(R) = cR^\alpha$ to

$$\sigma_\varepsilon^* = \frac{\sigma_\varepsilon}{Q(R)}\frac{N}{\alpha} \qquad \stackrel{\alpha=2}{\Longrightarrow} \qquad \sigma_\varepsilon^* = \frac{\sigma_\varepsilon N}{2cR^2}, \tag{60}$$

one can rewrite (57) by means of (58), (60), (41), and (28). We finally obtain

$$\frac{N}{R}\overline{x}_{1;\lambda} = \frac{c_{1,\lambda}\sigma^*}{\sqrt{1+(\sigma_\varepsilon^*/\sigma^*)^2 + \sigma^{*2}/2N}}. \tag{61}$$

5.1.4 *On the calculation of* $\overline{\mathbf{h}^2}_{1;\lambda}$

The derivation here follows the same ideas as in the previous subsection. Therefore, we will only sketch the way. The pdf of $\mathbf{h}^2_{1;\lambda}$ will be written as $p_{1;\lambda}(\mathbf{h}^2)$. By analogy to (43) and (45) we have

$$p_{1;\lambda}(\mathbf{h}^2) = \lambda p_{\mathbf{h}}(\mathbf{h}^2) P_{\mathrm{a}1,\lambda}(\mathbf{h}^2). \tag{62}$$

and

$$P_{\mathrm{a}1,\lambda}(\mathbf{h}^2) = \int p(\tilde{Q}|\mathbf{h}^2)[1-P_1(\tilde{Q})]^{\lambda-1}\, d\tilde{Q}. \tag{63}$$

The conditional pdf $p(\tilde{Q}|\mathbf{h}^2)$ is obtained from $p(\tilde{Q}|x,\mathbf{h}^2)$, Eq. (46), by integrating over all possible x-states with pdf $p_x(x)$, Eq. (44). This is analogous to (47)

$$p(\tilde{Q}|\mathbf{h}^2) = \int p(\tilde{Q}|x,\mathbf{h}^2)\, p_x(x)\, dx. \tag{64}$$

With (44), (46), and the substitution $t := x/\sigma$ the integral (64) becomes

$$p(\tilde{Q}|\mathbf{h}^2) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon}\frac{1}{\sqrt{2\pi}}\int \mathrm{e}^{-\frac{1}{2}t^2} \exp\left[-\frac{1}{2}\left(\frac{2cR\sigma}{\sigma_\varepsilon}t + \frac{\tilde{Q}-c(R^2+\mathbf{h}^2)}{\sigma_\varepsilon}\right)^2\right] dt. \tag{65}$$

Using the integration formula (39) and the auxiliary parameter $\tilde{\sigma}_{\mathbf{h}}$ (analogous to (52))

$$\tilde{\sigma}_{\mathbf{h}} := \sqrt{\sigma_\varepsilon^2 + (2cR\sigma)^2} \tag{66}$$

one gets

$$p(\tilde{Q}|\mathbf{h}^2) = \frac{1}{\sqrt{2\pi}\tilde{\sigma}_{\mathbf{h}}} \exp\left[-\frac{1}{2}\left(\frac{\tilde{Q} - cR^2 - c\,\mathbf{h}^2)}{\tilde{\sigma}_{\mathbf{h}}}\right)^2\right]. \tag{67}$$

Inserting into the acceptance probability expression (63), using the substitution $t := -[\tilde{Q} - c(R^2 + \sigma^2 N]/\tilde{\sigma}$ for the argument in (42), and applying the symmetry relation of the standard normal cdf $\Phi(t) = 1 - \Phi(-t)$, one obtains

$$P_{\mathrm{a}1,\lambda}(\mathbf{h}^2) = \frac{\tilde{\sigma}}{\tilde{\sigma}_{\mathbf{h}}} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} [\Phi(t)]^{\lambda-1} \exp\left[-\frac{1}{2}\left(\frac{-\tilde{\sigma}t + c\sigma^2 N - c\,\mathbf{h}^2}{\tilde{\sigma}_{\mathbf{h}}}\right)^2\right] dt. \tag{68}$$

Now, the expected value of $\mathbf{h}^2_{1;\lambda}$ can be calculated from (62)

$$\overline{\mathbf{h}^2}_{1;\lambda} = \lambda \int \mathbf{h}^2 p_{\mathbf{h}}(\mathbf{h}^2) P_{\mathrm{a}1,\lambda}(\mathbf{h}^2)\, d(\mathbf{h}^2). \tag{69}$$

With (48), (68), and writing u instead of $\mathbf{h}^2$, we have

$$\overline{\mathbf{h}^2}_{1;\lambda} = \frac{\lambda}{2\pi} \frac{\tilde{\sigma}}{\tilde{\sigma}_{\mathbf{h}}} \frac{1}{\sqrt{2N}\sigma^2} \int u \exp\left[-\frac{1}{2}\left(\frac{u - N\sigma^2}{\sqrt{2N}\sigma^2}\right)^2\right] \\ \times \int [\Phi(t)]^{\lambda-1} \exp\left[-\frac{1}{2}\left(\frac{cu + \tilde{\sigma}t - c\sigma^2 N}{\tilde{\sigma}_{\mathbf{h}}}\right)^2\right] dt\, du. \tag{70}$$

After the substitution $s := (u - N\sigma^2)/\sqrt{2N}\sigma^2$ and changing the integration order, we arrive at

$$\overline{\mathbf{h}^2}_{1;\lambda} = \frac{\lambda}{\sqrt{2\pi}} \int [\Phi(t)]^{\lambda-1} \frac{\tilde{\sigma}}{\tilde{\sigma}_{\mathbf{h}}} \frac{1}{\sqrt{2\pi}} \\ \times \int (N\sigma^2 + \sqrt{2N}\sigma^2 s)\, \mathrm{e}^{-\frac{1}{2}s^2} \exp\left[-\frac{1}{2}\left(\frac{c\sqrt{2N}\sigma^2}{\tilde{\sigma}_{\mathbf{h}}} s + \frac{\tilde{\sigma}}{\tilde{\sigma}_{\mathbf{h}}} t\right)^2\right] ds\, dt. \tag{71}$$

The inner integral can be treated by (39) and (56). Taking (66) and (52) into account one gets

$$\overline{\mathbf{h}^2}_{1;\lambda} = \frac{\lambda}{\sqrt{2\pi}} \int [\Phi(t)]^{\lambda-1} \left[N\sigma^2 \mathrm{e}^{-\frac{1}{2}t^2} - (\sqrt{2N}\sigma^2)^2 c \frac{t}{\tilde{\sigma}}\, \mathrm{e}^{-\frac{1}{2}t^2}\right] dt. \tag{72}$$

The first term in the sum of the integrand can be simplified, because $\frac{d}{dt}[\Phi(t)]^\lambda = \lambda[\Phi(t)]^{\lambda-1}\mathrm{e}^{-\frac{1}{2}t^2}/\sqrt{2\pi}$. Therefore, the integral yields $N\sigma^2$. The second term basically contains (58), i.e. it can be expressed by the progress coefficient $c_{1,\lambda}$

$$\overline{\mathbf{h}^2}_{1;\lambda} = N\sigma^2 \left[1 - \frac{2c\sigma^2}{\tilde{\sigma}} c_{1,\lambda}\right]. \tag{73}$$

After applying the normalization (28), (60), and (41) we finally obtain

$$\frac{\overline{\mathbf{h}^2}_{1;\lambda}}{R^2} = \frac{\sigma^{*2}}{N}\left[1 - \frac{1}{N}\frac{c_{1,\lambda}\sigma^*}{\sqrt{1+(\sigma^*_\varepsilon/\sigma^*)^2 + \sigma^{*2}/2N}}\right]. \tag{74}$$

5.1.5 *Putting things together*

The progress rate formula is obtained by plugging (74) and (61) in (30). In order to make it more comparable to former progress rate formulae it will be further simplified by approximation of the square root in (30) using Taylor expansions. For sake of simplicity, we use the abbreviations

$$a := 1 + \frac{\sigma^{*2}}{\kappa^2 N} \qquad \text{and} \qquad b := \sqrt{1+(\sigma^*_\varepsilon/\sigma^*)^2 + \sigma^{*2}/2N}. \tag{75}$$

The progress rate φ^* reads

$$\varphi^*_{1,\lambda} = N\left[1 - \sqrt{\left(1 - \frac{1}{\kappa N}\frac{c_{1,\lambda}\sigma^*}{b}\right)^2 + \frac{\sigma^{*2}}{\kappa^2 N} - \frac{c_{1,\lambda}\sigma^*}{\kappa N}\frac{\sigma^{*2}}{\kappa N}\frac{1}{b}}\right] + \dots. \tag{76}$$

Since $c_{1,\lambda}\sigma^*/b$ has the upper bound $c_{1,\lambda}\sqrt{2N}$, the quadratic part of $(\cdots)^2$ in (76) can be neglected for sufficiently large κN. Thus, one obtains

$$\begin{aligned}\varphi^*_{1,\lambda} &= N\left[1 - \sqrt{a - 2\frac{1}{\kappa N}\frac{c_{1,\lambda}\sigma^*}{\kappa N}\frac{1}{b} - \frac{c_{1,\lambda}\sigma^*}{\kappa N}\frac{\sigma^{*2}}{\kappa N}\frac{1}{b}}\right] + \dots \\ \varphi^*_{1,\lambda} &= N\left[1 - \sqrt{a}\sqrt{1 - 2\frac{c_{1,\lambda}\sigma^*}{\kappa N}\frac{1}{ab}\left(1 + \frac{\sigma^{*2}}{2\kappa N}\right)}\right] + \dots.\end{aligned} \tag{77}$$

With the Taylor expansion $\sqrt{1-2x} = 1 - x + \mathcal{O}(x^2)$ one gets

$$\varphi^*_{1,\lambda} = N\left[1 - \sqrt{a} + \frac{c_{1,\lambda}\sigma^*}{\kappa N}\frac{1}{\sqrt{a}\,b}\left(1 + \frac{\sigma^{*2}}{2\kappa N}\right)\right] + \dots \tag{78}$$

which gives with (75) the final result

$$\boxed{\varphi^*_{1,\lambda} = \frac{1}{\kappa}\left[c_{1,\lambda}\sigma^* \frac{1+\frac{\sigma^{*2}}{2\kappa N}}{\sqrt{1+\frac{\sigma^{*2}}{\kappa^2 N}}\sqrt{1+\left(\frac{\sigma^*_\varepsilon}{\sigma^*}\right)^2+\frac{\sigma^{*2}}{2N}}} - N\left(\sqrt{\kappa^2+\frac{\sigma^{*2}}{N}}-\kappa\right)\right].} \tag{79}$$

This progress rate formula contains special cases already calculated before: For $\kappa = 1$ and $\sigma^*_\varepsilon = 0$ one gets the N dependent noise-free φ formula derived in [25, p. 385]. For $\sigma^*_\varepsilon > 0$ and $N \to \infty$ ($\sigma^* < \infty$) one gets the asymptotic noisy φ formula [32], it reads

$$\varphi^*_{1,\lambda} \simeq \frac{1}{\kappa}\left[\frac{c_{1,\lambda}\sigma^*}{\sqrt{1+(\sigma^*_\varepsilon/\sigma^*)^2}} - \frac{\sigma^{*2}}{2\kappa}\right]. \tag{80}$$

In order to see the influence of N on φ^*, given a fixed normalized noise strength σ^*_ε, contour plots of $\varphi^*(\sigma^*, \kappa)$ are presented for the $(1,10)$-ES in Figure 4. As one can infer from the plots, as long as $N < \infty$ there is always an optimum $(\sigma^*, \kappa) = (\hat{\sigma}^*, \hat{\kappa})$ combination that maximizes the expected progress φ^*. With increasing N, the optimum shifts to larger $\hat{\sigma}^*$ and $\hat{\kappa}$. Actually, one can show for $N \to \infty$ that $\hat{\sigma}^*$, $\hat{\kappa} \to \infty$ and the progress rate maximum becomes $c^2_{1,\lambda}/2$ [32]. That is, the effect of the noise would be totally removed ($c^2_{1,\lambda}/2$ is the maximum progress rate of the noise-free case). We will come back to the related convergence properties in Section 5.3.

It is interesting to compare the asymptotic result (80) for $\kappa = 1$ with results of Miller and Goldberg (1997) [15] in the GA field. Their work is similar to the standard approach of ES theory: The GA is treated as a dynamical system. One is basically interested in the generational change of this system. They investigated the expected change of the fitness values, known as *quality gain* $\overline{Q}$ in ES theory [33]

$$\overline{Q} := \mathrm{E}\{\langle F\rangle^{(g+1)} - \langle F\rangle^{(g)}\}. \tag{81}$$

Assuming a normally distributed population fitness, Miller and Goldberg obtained

$$\overline{Q} := I\,\sigma_F^{2(g)} \Big/ \sqrt{\sigma_F^{2(g)} + \sigma^2_\varepsilon}, \tag{82}$$

where σ^2_F is the population variance of the fitness and I is the so-called selection intensity adopted from quantitative genetics (see e.g. Blumer [34]). For

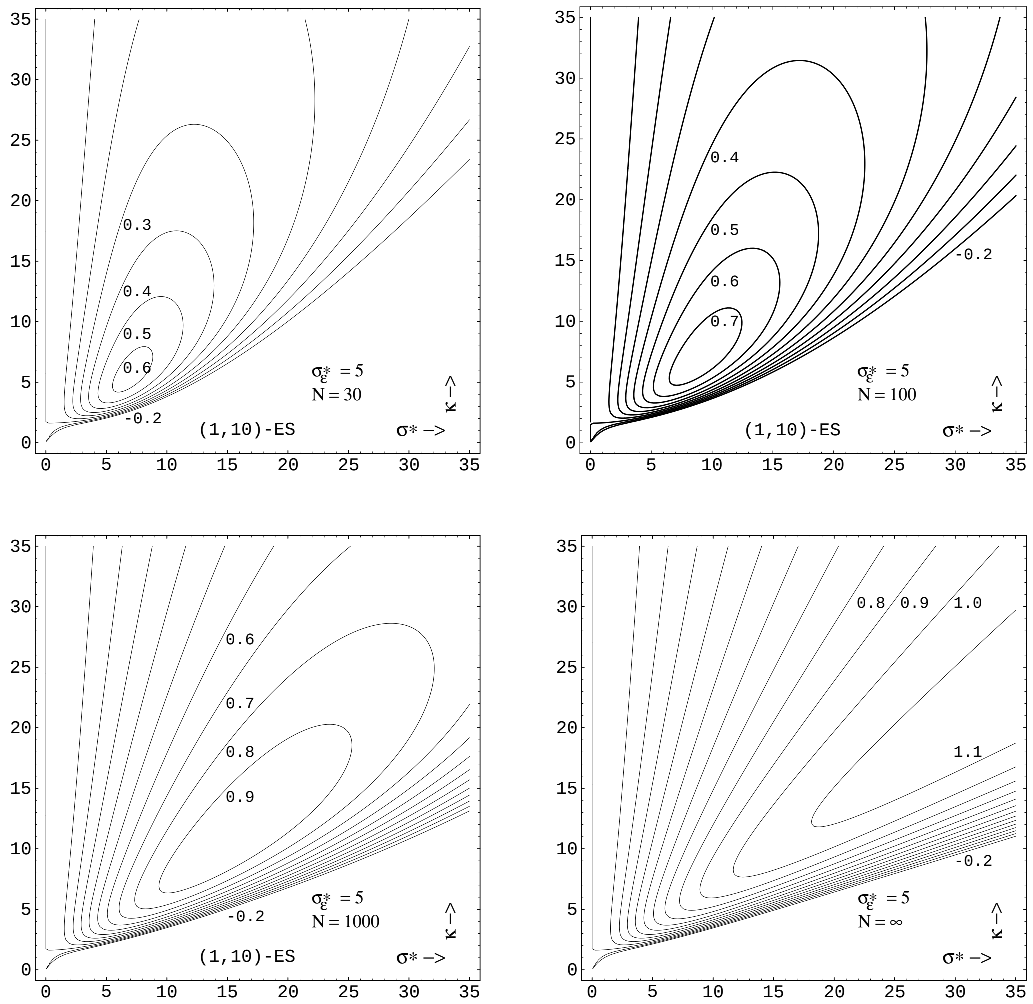

Fig. 4. On the influence of N on φ^*. The $(1, 10)$-ES is considered with $\sigma_\varepsilon^* = 5$. The contour lines are plots of constant φ^* values.

$(1, \lambda)$ selection one has $I = c_{1,\lambda}$ with $c_{1,\lambda}$ defined by (58). Equation (82) can be compared with the asymptotic quality gain formula of the $(1, \lambda)$-ES on the sphere model. Using results from [2] one finds $\overline{Q} \simeq -\varphi^* \alpha Q/N = \text{const.} \cdot \varphi^*$. Comparing this with (82) taking (80) into account, one sees that the functional structure of the gain term in (80) is recovered if the population variance σ_F^2 of the GA is equal to the (appropriately normalized) mutation strength in the $(1, \lambda)$-ES. This is a remarkable observation because the whole population variance in $(1, \lambda)$-ESs is produced by applying the mutation operator to a single parent.

The only significant difference between (82) and (80) is due to the loss term in (80). It reflects the detrimental effect of the mutations which can be neglected when the mutation strength is sufficiently small. At the other hand one may speculate on the validity domain of (82). Due to the normal approximation approach used, deviations are to be expected. First, the model of Miller does consider selection only. That is, the effects of recombination and/or mutation are not explicitly covered by his analysis. Second, the normal distribution assumption will be more and more violated by the successive approach of the GA to the optimum because the noise free fitness values are bounded by the optimum. That is, the skewness of the fitness distribution cannot be neglected further and correction terms are to be incorporated in (82).

5.2 Comparison with experiments

In [32] it has been shown that the asymptotic φ^* formula (80) gives satisfactory results even for small N, such as $N = 30$, as long as σ^* and κ are not too large. However, as one can see from Figure 4, σ^* and κ should be chosen relatively large in order to have a high convergence velocity. Therefore, an $(1, 10)$-ES example with $\sigma^*_\varepsilon = 5$ and $\kappa = 10$ will be considered which demonstrates the limitations of the asymptotic theory ($N \to \infty$). Figure 5 shows results from so-called "one-generation experiments". That is, for a fixed mutation strength

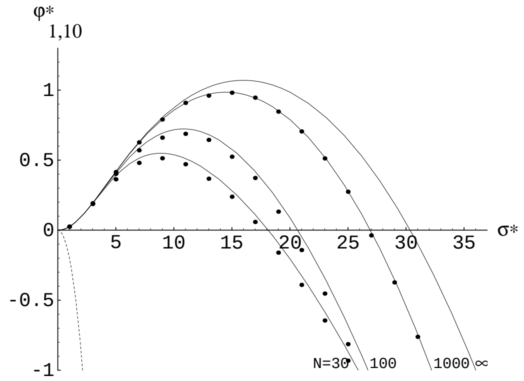

Fig. 5. Simulations of the $(1,10)$-ES with $\sigma^*_\varepsilon = 5$ and $\kappa = 10$ are displayed by dots for parameter space dimensions $N = 30$, 100, and 1000. The noisy fitness model $\tilde{Q} = \|\mathbf{y}\|^2 + \mathcal{N}(0, \sigma_\varepsilon^2)$ with $\sigma_\varepsilon = 2\|\mathbf{y}_\mathrm{p}\|^2\sigma^*_\varepsilon/N$ has been used. The dashed curve shows φ^* for $\kappa = 1$, i.e. the standard ES without rescaling.

σ the parent $\mathbf{y}_\mathrm{p}$ is randomly placed at a distance R to the optimum. Then, algorithm (20) is applied to one generation and the parental distance change serves as an estimate for $\varphi(\sigma)$. As one can see, the curves provide satisfactory predictions for the data points simulated. Furthermore, it should be clear that the asymptotic φ^* formula (80), i.e. the $N = \infty$ curve, is not well suited for the parameter setting used.

The derivations presented in this paper are for the fitness model $Q(R) = cR^\alpha$ with $\alpha = 2$. However, just like the noisy case *without* rescaled mutations [2] it seems possible to derive an approximate φ^* formula for $\alpha \neq 2$. On the lowest level of approximation φ^* is almost unchanged. One simply has to change the σ^*_ε normalization rule according to (60). Figure 6 shows results from $(1, 10)$-ES simulations using the same setting as in Figure 5; the left picture displays the

$\alpha = 1$ case and the right one $\alpha = 4$. As one can see, the φ^* formula (79) can be

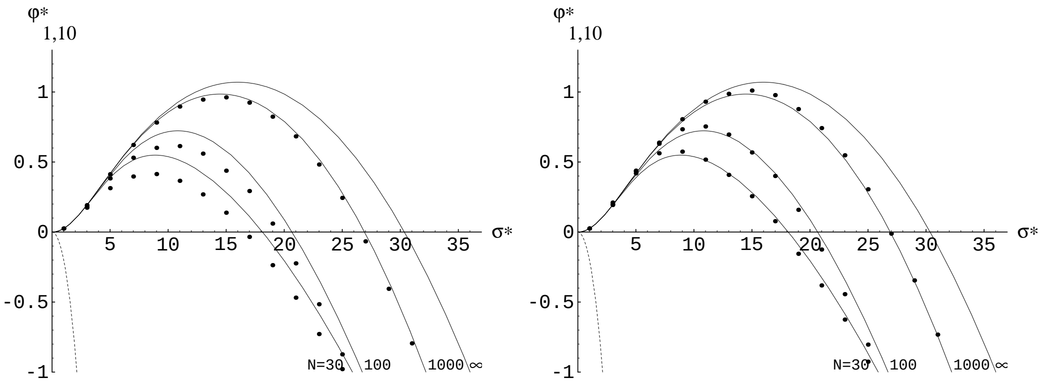

Fig. 6. Left picture: Fitness model $\tilde{Q} = \|\mathbf{y}\| + \mathcal{N}(0, \sigma_\varepsilon^2)$ with $\sigma_\varepsilon = \|\mathbf{y}_\mathrm{p}\|\sigma_\varepsilon^*/N$. Right picture: Fitness model $\tilde{Q} = \|\mathbf{y}\|^4 + \mathcal{N}(0, \sigma_\varepsilon^2)$ with $\sigma_\varepsilon = 4\|\mathbf{y}_\mathrm{p}\|^4\sigma_\varepsilon^*/N$. See also Figure 5.

applied to $\alpha \neq 0$, however, one has generally to expect larger approximation errors.

5.3 On the convergence properties of the rescaled mutation technique

Depending on the choice of the strategy parameters σ, κ, λ, and the noise level σ_ε the $(1, \lambda)$-ES exhibits different convergence behavior. We will discuss different scenarios and derive an evolution criterion which predicts the general behavior of the ES.

The progress rate φ is the key for understanding the convergence properties because it measures the generational distance change $\varphi = R^{(g)} - R^{(g+1)}$. With the normalization (28) one obtains

$$R^{(g+1)} = R^{(g)} \left(1 - \frac{\varphi^*(\sigma^*, \sigma_\varepsilon^*, \kappa, \lambda, N)}{N}\right). \tag{83}$$

This difference equation governs the mean value dynamics of the ES. Maximal performance, i.e. the fastest R decrease, is obtained for maximal φ^*. As we have seen in Figure 4, given fixed λ, N, and σ_ε^* there is an optimum (σ^*, κ) choice. However, the question arises whether this is a real-world scenario. That is:

(1) Can the algorithm be expected to be in a nearly optimum state $\sigma^* \approx \hat{\sigma}^*$?
(2) Is $\sigma_\varepsilon^* \approx$ const. a realistic assumption?

The answer to the first question shall be postponed to the next section. As to the second one, we recall definition (59) $\sigma_\varepsilon^* = \sigma_\varepsilon N/|Q'|R \stackrel{!}{=}$ const. That is, σ_ε must be proportional to $|Q'|R$. The so defined $\sigma_\varepsilon(R)$ function becomes interpretable when $Q(R) = Q'R/\alpha$ is postulated, where $\alpha > 0$ is a constant. The differential equation $Q(R) = Q'R/\alpha$ can be solved for Q yielding $Q = cR^\alpha$. With this fitness function, the ratio σ_ε/Q can be interpreted as the *relative* measurement error

$$\varepsilon_r := \frac{\sigma_\varepsilon(R)}{Q(R)} \stackrel{!}{=} \text{const.} \tag{84}$$

and $\sigma_\varepsilon^* = \varepsilon_r N/\alpha \neq f(R)$ is independent of R. Such a fitness model has the peculiarity that depending on λ and N, given a fixed κ, there is a relative measurement error $\hat{\varepsilon}_r = \sigma_\varepsilon^* \alpha/N$ above which the ES cannot converge at all. $\hat{\varepsilon}_r$ can be calculated by means of the *evolution criterion* to be introduced below.

Besides $\sigma_\varepsilon^* = \text{const.}$, the case $\sigma_\varepsilon = \text{const.}$ is of practical interest. It was one of the main subjects of Section 3. As we have seen, $\sigma_\varepsilon = \text{const.}$ implies convergence to a *residual distance* R_∞. The derivation of a lower bound for R_∞ will be prepared now.

In Figure 4, the progress rate φ^* has been displayed as a function of σ^* and κ. Since for convergence $R^{(g+1)} < R^{(g)}$ and divergence $R^{(g+1)} > R^{(g)}$ does hold, the limit case $R^{(g+1)} = R^{(g)}$ yields with (83)

$$R^{(g+1)} = R^{(g)} =: R_\infty \qquad \Longleftrightarrow \qquad \varphi^*(\sigma^*, \sigma_\varepsilon^*, \kappa, \lambda, N) = 0. \tag{85}$$

That is, the curve $\varphi^* = f(\sigma^*, \sigma_\varepsilon^*) = 0$. Figure 7 shows such curves obtained from (79) for the $(1,10)$-ES taking κ and N as parameters.

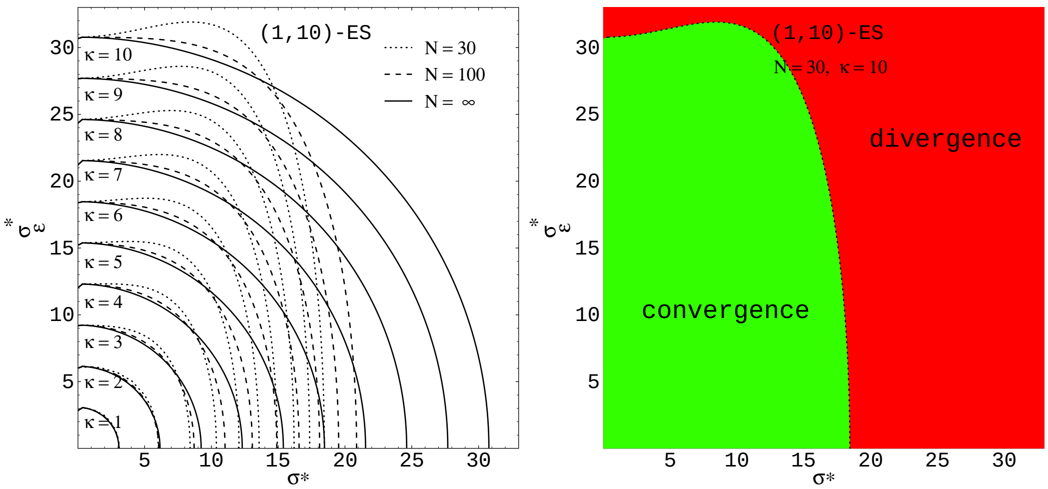

Fig. 7. The convergence card of the $(1,10)$-ES with rescaled mutation. The right picture explains how to "read" the curves in the left one. Each curve belongs to a special setting of N and κ.

The states $(\sigma^*, \sigma_\varepsilon^*)$ for which the ES is not divergent can be calculated as well. Since for non-divergence $\varphi^* \geq 0$ must hold, Eq. (77) yields after a simple calculation the

$$\boxed{\underline{\text{Evolution Criterion:}} \qquad 4c_{1,\lambda}^2 \geq \frac{\sigma^{*2} + \sigma_\varepsilon^{*2} + \sigma^{*4}/2N}{(\kappa + \sigma^{*2}/2N)^2}.} \tag{86}$$

As one can see, given *bounded values* of σ^* and σ_ε^*, convergence (i.e., the ">" sign) can always be ensured either by increasing the population size λ (since $c_{1,\lambda} \sim \sqrt{2 \ln \lambda}$) or by increasing κ. The latter is the recommended measure because it does not require additional fitness calculations.

The evolution criterion allows for a calculation of the residual distance R_∞, when $\sigma_\varepsilon = \text{const.}$ is given. Since R_∞ refers to the condition (85), the equal sign in (86) is to consider. Resolving (86) for σ_ε^* gives

$$\sigma_\varepsilon^* = 2c_{1,\lambda}\kappa \sqrt{\left(1 + \frac{\sigma^{*2}}{2\kappa N}\right)^2 - \left(\frac{\sigma^*}{2\kappa c_{1,\lambda}}\right)^2 \left(1 + \frac{\sigma^{*2}}{2N}\right)}. \tag{87}$$

The σ_ε^* definition (59) is applied to the fitness model $Q = cR^\alpha$ at $R = R_\infty$. This gives $\sigma_\varepsilon^* = \frac{\sigma_\varepsilon N}{\alpha} \Big/ cR_\infty^\alpha$; equating to (87), one can solve for R_∞

$$R_\infty = \sqrt[\alpha]{\frac{\sigma_\varepsilon N}{2c\alpha\kappa c_{1,\lambda}} \left[\left(1 + \frac{\sigma^{*2}}{2\kappa N}\right)^2 - \left(\frac{\sigma^*}{2\kappa c_{1,\lambda}}\right)^2 \left(1 + \frac{\sigma^{*2}}{2N}\right)\right]^{-\frac{1}{2}}}. \tag{88}$$

If one neglects $\sigma^{*2}/2\kappa N$, Eq. (88) can be turned in an inequality because the negative term in (88) increases R_∞. Thus, we finally get the simple expression

$$\boxed{R_\infty \geq \sqrt[\alpha]{\frac{\sigma_\varepsilon N}{2c\alpha\kappa c_{1,\lambda}}}} \tag{89}$$

where the equal sign holds for vanishing σ^*. Note, Eq. (89) contains (12) as a special case. Obviously, by increasing κ the $(1, \lambda)$-ES with rescaled mutations defined by (21) allows for an arbitrary reduction of the residual distance R_∞.

6 How to take advantage of the rescaling technique

After the excursion into ES theory, we are now prepared for the application side. As we have learned from theory, the rescaling technique (20) allows for an reduction of the residual distance (89). Furthermore, if one is able to control the mutation strength appropriately, the detrimental effect of noise on the convergence velocity (progress rate φ) should be reduced. However, in order to take advantage of the rescaling technique, the self-adaptation (SA) part of the algorithms must work in such a way that the ES dynamically learns the right mutation strength.

Up to now there is no theoretical analysis on the full-blown κ-$(1, \lambda)$-SA-ES. Therefore, one has to rely on simulation experiments. In the following first subsection, the standard σ-SA is evaluated. It will be shown that this σ-SA has considerable problems to learn the right mutation strength. Therefore, advanced σ-SA rules will be proposed and evaluated in Section 6.2 and 6.3.

6.1 On the performance of the standard σ-self-adaptation rule

In order to take advantage of the rescaling technique (21), the progress rate φ^* should be nearly maximal. When looking at the Figures 4–6, one gets the feeling that optimal performance is obtained at normalized mutation strengths $\hat{\sigma}^*$ which are much larger than the $\hat{\sigma}^*$ of the standard $(1, \lambda)$-ES where $\hat{\sigma}^* = c_{1,\lambda}$ (for $N \to \infty$) holds. Actually, in [32] an approximative $\hat{\sigma}^*$ formula has been derived that gives for $N \to \infty$ and $\sigma^*_\varepsilon \ll \kappa c_{1,\lambda}$ an optimum $\hat{\sigma}^* \approx \kappa c_{1,\lambda}$.

It is the duty of the SA (self-adaptation) part of the real algorithm (see e.g. (9), second line) to change the mutation strength σ in such a way that for each generation $\sigma^* \approx \hat{\sigma}^*$ is roughly fulfilled. Standard SA techniques, as implemented in algorithm (9), seem to work well for $(1, \lambda)$-ESs as long as the fitness is noise free. Even in the noisy case with $\kappa = 1$, algorithm (9) works well. It realizes R_∞ given by (12) as one can see in the upper two pictures of Figure 2. Unfortunately, things change to the worse when rescaled mutations *and* noisy fitness come into play.

As an example algorithm (23) is tested with $\kappa = 50$ on a $(1, 60)$-ES with $N = 10$, $\sigma_\varepsilon = 1.0$ and with $N = 100$, $\sigma_\varepsilon = 0.1$. Figure 8 shows the $\langle R \rangle$ dynamics (upper pictures) and the average of the normalized mutation strength σ^* (lower picture). The case we are dealing with is labeled by "`SA: LN, standard`".

For the case $N = 10$ it seems that the ES converges, however, very slowly. When looking at σ^* (lower left picture) we see that σ^* is far too small. Actually, it is even considerably smaller than the noise free case ($\sigma_\varepsilon = 0$, $\kappa = 1$, displayed

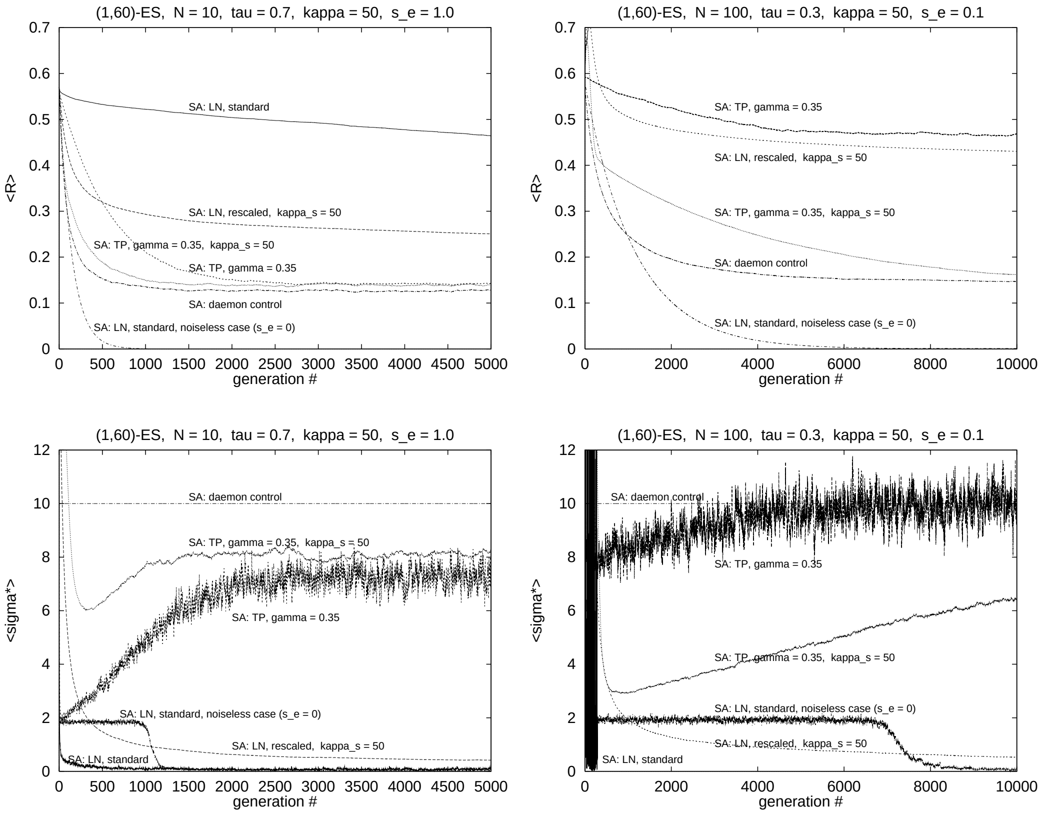

Fig. 8. Evolution dynamics of various SA techniques on a $(1,60)$-ES. The left pictures are for parameter space dimension $N = 10$ and noise strength $\sigma_\varepsilon = \text{s\_e} = 1.0$. The curves are obtained by averaging over 300 independent ES runs. The right pictures are for $N = 100$, $\sigma_\varepsilon = \text{s\_e} = 0.1$ obtained by averaging over 100 ES runs. The upper pictures display the $\langle R \rangle$ dynamics, whereas the lower two display the average of the normalized mutation strength $\sigma^* =\, <\text{sigma}^*>$.

as "`SA: LN, standard, noiseless case (s_e = 0)`") where the optimum $\hat{\sigma}^* = c_{1,60} \approx 2.32$ is roughly reached. However, adapting smaller σ^* values as desired is not a general tendency. For $N = 100$, one even observes divergence. The lower right picture shows extreme σ^* fluctuations which fill up the picture to the left. And $\langle R \rangle$ leaves the plotting interval after a few ten generations.

One might wonder whether the rescaled mutation technique can reach the theoretical R_∞ values at all. It is an advantage of the sphere model that, given a desired σ^*, one can calculate the mutation strength $\sigma^{(g)}$ needed. Because of the normalization (28) it holds $\sigma^{(g)} = \sigma^* R^{(g)}/N$. A "daemon" who "knows" $R^{(g)}$ could tune $\sigma^{(g)}$ deterministically. Since the distance to the optimum is known for the sphere model, the "daemon's task" can be easily implemented. The dynamics obtained by this control is also depicted in Figure 8 labeled as "`SA: daemon control`". A $\sigma^* = 10$ has been chosen. As one can verify by means of Eq. (89) $R_\infty \approx 0.147$ is indeed obtained for $N = 100$. For $N = 10$ the actually measured R_∞ is even a bit below the theoretical limit. As we can clearly see, the rescaling technique works. It is obvious, the problems arise from the SA mechanism that

(a) produces too small σ^* values and
(b) exhibits too large σ^* fluctuations that disturb the evolution process.

Item (b) needs further discussion to be done below.

6.2 Advanced σSA rule: Rescaled σ mutations

In order to understand the σ^* dynamics observed from the standard σSA, we have to remember that the actual parental σ^* is inherited from the offspring owning the best fitness value. The rescaling, however, takes place *after* the selection. There is *no* feedback from the fitness of the parent actually produced after the rescaling. In other words, the σSA *can only* learn the optimum σ^* *without* rescaling. In the author's opinion, there is no hope to find a solution to this problem that does not radically change the basic SA idea. Of course, one might imagine meta-ESs and aging mechanism; however, these mechanisms violate the time locality of the σSA algorithm. That is, for such strategies the stochastic behavior of the EA is not fully determined by the parental state, but by a *history of states* dating back more than one generation (Markov process of higher order). Though this might open up interesting new σ control techniques, we will stick here to the possibilities inherent in the frame of time locality.

There are three proposals to improve the standard σSA algorithm. The first – originated by Rechenberg [22] – transfers the idea of rescaled mutations from the object parameters to the strategy parameters. Since the mutation of σ is done by multiplication, see line 2 in algorithms (9) or (23), a full analogy can be drawn when expressed by the logarithm law. In case of the $(1, \lambda)$-ES, the parental $\sigma_{\rm p}^{(g+1)}$ is obtained from $\sigma_{\rm p}^{(g)}$ by that random multiplicator which belongs to the best offspring: $\sigma_{\rm p}^{(g+1)} := \sigma_{\rm p}^{(g)} \cdot \xi_{1;\lambda}^{(g)} \;\Rightarrow\; \ln \sigma_{\rm p}^{(g+1)} := \ln \sigma_{\rm p}^{(g)} + \ln \xi_{1;\lambda}^{(g)}$. Therefore, the rescaling rule reads

$$\ln \sigma_{\rm p}^{(g+1)} := \ln \sigma_{\rm p}^{(g)} + \frac{1}{\kappa_\sigma} \ln \xi_{1;\lambda}^{(g)}, \qquad \kappa_\sigma \geq 1, \tag{90}$$

which is in analogy to (20). As in Section 3.2, Eq. (90) can be expressed by the best offspring and its corresponding mutation. The comparison with (22) yields

$$\ln \sigma_l^{(g+1)} := \ln \sigma_{1;\lambda}^{(g)} + \left(\frac{1}{\kappa_\sigma} - 1\right) \ln \xi_{1;\lambda}^{(g)} + \ln \xi_l^{(g+1)}. \tag{91}$$

Now we can write down the full κ-κ_σ-σSA-$(1,\lambda)$-ES

$$\forall\, l = 1\ldots\lambda \;:\; \begin{cases} \xi_l^{(g+1)} := \Xi(\tau, \kappa_\sigma, \ldots) \\ \sigma_l^{(g+1)} := \sigma_{1;\lambda}^{(g)} \left(\xi_{1;\lambda}^{(g)}\right)^{\frac{1}{\kappa_\sigma}-1} \xi_l^{(g+1)} \\ \mathbf{z}_l^{(g+1)} := \sigma_l^{(g+1)} \vec{\mathcal{N}}(0,1) \\ \mathbf{y}_l^{(g+1)} := \mathbf{y}_{1;\lambda}^{(g)} + \left(\frac{1}{\kappa} - 1\right) \mathbf{z}_{1;\lambda}^{(g)} + \mathbf{z}_l^{(g+1)}. \end{cases} \tag{92}$$

The first line in (92) has been modified compared to (23). Ξ stands for a random number generator which produces ξ. In the standard σSA the log-normal generator $\exp[\tau\mathcal{N}(0,1)]$ has been used. Another type of generator – the two-point rule – will be discussed below.

As to the σ-rescaling, Rechenberg [22, p. 197] proposed the use of $\Xi(\tau,\kappa_\sigma) = [\exp(\tau\mathcal{N}(0,1))]^{\kappa_\sigma}$ which is basically a proportional scaling of the learning parameter $\tau \mapsto \kappa_\sigma\tau$. The intention is to produce larger σ-changes that should allow for a more reliable detection of the direction of σ-change. After selection, the most promising direction is inherited rescaled according to (91). Experiments performed by the author indicate, however, that the main effect of (91) is *smoothing* of the stochastic σ-dynamics. This effect can be observed in the lower right picture of Figure 8. The curve with the label "`SA: LN, rescaled, kappa_s = 50`" is astonishing smooth, especially when compared to "`SA: LN, standard`". However, the $\langle R\rangle$ performance is rather slow for both $N = 10$ and $N = 100$. The convergence velocity becomes even worse when the initial $\sigma^{(0)}$ is chosen too small. That is, the κ_σ-rescaling slows down the SA dynamics.

6.3 Advanced σSA rules: Unsymmetric two-point mutations and its hybrid with κ_s-rescaling

There is a second mutation rule occasionally used in ESs – the *symmetric two-point* rule proposed by Rechenberg [22, p. 47]. It is a special case of the general two-point rule introduced in [24] which reads

$$\Xi(\tau,\gamma) := \begin{cases} 1+\tau, & \text{if } \mathrm{u}(0,1] > \gamma \\ 1/(1+\tau), & \text{if } \mathrm{u}(0,1] \le \gamma \end{cases}, \qquad \tau \gtrapprox \frac{c_{1,\lambda}}{\sqrt{N}}, \tag{93}$$

where $\mathrm{u}(0,1]$ is a sample from a uniform random number generator with $\mathrm{u} \in (0,1]$. The symmetrical case is obtained for $\gamma = 1/2$.

Since the symmetrical two-point rule exhibits the same behavior as the log-normal rule – the stationary σ^* values are too small – the author's idea [32] is to bias the mutations toward a σ increase. That is, the probability of increasing σ should be larger than 1/2. This is accomplished by choosing $\gamma < 1/2$. The evolution dynamics produced by the unsymmetric two-point rule is displayed in Figure 8 for $\gamma = 0.35$. In order to separate the different effects the σ-rescaling is switched off ($\kappa_\sigma = 1$); the curves obtained are labeled "`SA: TP, gamma = 0.35`". For the $N = 10$ case, one observes an $\langle R \rangle$ dynamics which approaches the vicinity of the steady state R_∞ within 3000 generations. Looking at the σ^* dynamics, one sees that the steady state distribution of σ^* has a much higher average compared to those of the mutation rules discussed before. However, this is brought at the expense of very large σ fluctuations: Due to the averaging over 300 ($N = 10$, left pictures in Figure 8) and 100 (right pictures) independent evolution runs, the real fluctuations are by a factor of $\sqrt{300} \approx 17$ and $\sqrt{100} = 10$, respectively, larger than displayed in Figure 8. Even though the average σ^* may be at its optimum $\hat{\sigma}^*$, the σ^* fluctuations must *necessarily* deteriorate the convergence velocity: Due to the general property of the φ^* curves (see e.g. Figure 5) there is only a bounded σ^* interval $(0, \sigma_0^*)$ for which $\varphi^* > 0$. Fluctuations outside this interval contribute with *negative* progress to the net progress; the performance degrades and the convergence velocity slows down (or even turns to divergence). Actually, this is observed in the upper right picture of Figure 8. The curve "`SA: TP, gamma = 0.35`" exhibits a slow and unsteady convergence behavior although the average σ^* is roughly at 10. Neglecting the fluctuations, one would expect a performance which is comparable to the "`SA: daemon control`" curves (where $\sigma^* = 10$ was chosen)!

As one can see, using $\gamma < 1/2$ alone provides larger σ^* values; however, due to the large σ^* fluctuations this advantage can get lost. Therefore, it should be the goal to reduce the fluctuation variance *without* changing the mean too much. This can be accomplished by the κ_σ-rescaling technique. That is, algorithm (92) is used with unsymmetric two-point mutations *and* $\kappa_\sigma > 1$. This hybrid performs considerably better than the variants presented before. The performance curves are labeled "`SA: TP, gamma = 0.35, kappa_s = 50`" in Figure 8. The $N = 10$ $\langle R \rangle$ curve comes close to the "daemon" curve. The $N = 100$ case shows a slightly different behavior, indicating the existence of two time scales which are not fully understood up to now. However, even in this case the hybrid σSA technique enables the rescaled mutation technique (21) to reach its theoretically predicted R_∞ value. Therefore, it is recommended for further investigations. A still open question concerns the choice of γ and κ_σ. No answer can be given here – we have reached the frontiers of ES research.

7 Summary and Outlook

Optimization in noisy and uncertain environments is regarded as one of the favorite application domains of evolutionary algorithms. Compared to its practical relevance, the effects of noise and its influence on the performance of the EAs have gained only little attention in EA research.

Research in the field of noisy EAs is still in its infancy. That is why this paper focussed on different topics. It first gave an short overview over the research that has been done in the established EA classes GA, EP (evolutionary programming), and ES up to now. The second goal was to show peculiarities that arise when noise deceives the selection process. Perhaps it may have appeared as a surprise, but the effects of fitness noise seems to be similar in GAs and ESs:

(a) reduction of convergence velocity, and
(b) deterioration of the final optimum location quality ($R_\infty > 0$).

Although this has been tested and quantified for the sphere model only, it should have been clear that the effects are of universal nature. However, its quantification on simple GA test functions, such as OneMax, still remains to be done. Anyway, the detoriative effects are present in all kinds of EAs. Therefore, it is of importance that the practitioners become aware of these facts and the methods for improving the convergence properties.

The main convergence improvement techniques, resampling and population up-sizing have been identified, however, there is still a need for an explicite EA theory giving answers as to the sizing of the population. As to evolution strategies (ESs) it is important that recombination is recommended. However, in real-valued search spaces, mutation rescaling techniques may be used alternatively. This technique might work for integer search spaces, too, but it has not been tested yet.

As to the ES theory discussed for the $(1, \lambda)$-ES with rescaled mutations in Section 5 and 6, there are two main directions which should be investigated:

- Development of a corresponding theory for (μ, λ)- and $(\mu/\mu, \lambda)$-ESs.
- Analysis of σSA rules with $\kappa_\sigma > 1$.

The derivation of the R_∞ formulae (14) and (15) must be substantiated by a respective progress rate theory. Such a theory would provide a deeper insight into the convergence improving mechanism taking place in populations.

A rather badly understood issue of ES theory concerns the self-adaptation (SA). As we have seen, the performance of the *real* algorithm is mainly deteriorated through maladjustments and large-scale fluctuations of the endogenous strategy parameters. Therefore, the investigation of SA techniques either analytically or by *clever* experiments should be given a high priority, because the techniques currently used need to be improved. It is the author's hope that this paper will initiate further investigations in this direction.

Acknowledgments

This paper is a result of research performed by the author as Heisenberg fellow of the Deutsche Forschungsgemeinschaft (DFG), under grant Be 1578/4-1. The author is grateful to Kalyanmoy Deb and A. Irfan Oyman for helpful comments. He also thanks the editors of the special edition for inviting him to contribute to this CMAME issue.

References

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

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

[3] J.M. Fitzpatrick and J.J. Grefenstette. Genetic Algorithms in Noisy Environments. In P. Langley, editor, *Machine Learning: Special Issue on Genetic Algorithms*, volume 3, pages 101–120. Kluwer Academic Publishers, Dordrecht, 1988.

[4] U. Hammel and T. Bäck. Evolution Strategies on Noisy Functions. How to Improve Convergence Properties. In Y. Davidor, R. Männer, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature, 3*, pages 159–168, Heidelberg, 1994. Springer-Verlag.

[5] T. Bäck and U. Hammel. Evolution strategies applied to perturbed objective functions. In Z. Michalewicz, J. D. Schaffer, H.-P. Schwefel, D. B. Fogel, and H. Kitano, editors, *Proc. First IEEE Conf. Evolutionary Computation*, pages 40–45. IEEE Press, Piscataway NJ, 1994.

[6] P. J. Angeline. The Effects of Noise on Self-Adaptive Evolutionary Optimization. In L. J. Fogel, P. J. Angeline, and T. Bäck, editors, *Proceedings of the Fifth Annual Conference on Evolutionary Programming*, pages 432–439. The MIT Press, Cambridge, MA, 1996.

[7] D. B. Fogel. *Evolutionary Computation.* IEEE Press, New York, 1995.

[8] B. Levitan and S. Kauffman. Adaptive walks with noisy fitness measurements. *Molecular Diversity*, 1(1):53–68, 1995.

[9] S. Rana, L. D. Whitley, and R. Cogswell. Searching in the Presence of Noise. In H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature, 4*, pages 198–207, Heidelberg, 1996. Springer.

[10] D. E. Goldberg and M. W. Rudnick. Genetic Algorithms and the Variance of Fitness. *Complex Systems*, 5(3):265–278, 1991.

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

[12] D. E. Goldberg, K. Deb, and J. H. Clark. Genetic Algorithms, Noise, and the Sizing of Populations. *Complex Systems*, 6(4):333–362, 1992.

[13] G. R. Harik, E. Cantú-Paz, B. L. Miller, and D. E. Goldberg. The Gambler's Ruin Problem, Genetic Algorithms, and the Sizing of Populations. In *Proceedings of 1997 IEEE Int'l Conf. on Evolutionary Computation (ICEC '97)*, pages 7–12, Indianapolis IN, 1997. IEEE Press, Piscataway NJ.

[14] B. L. Miller. *Noise, Sampling, and Efficient Genetic Algorithms.* PhD thesis, University of Illinois at Urbana-Champaign, Urbana, IL 61801, 1997. IlliGAL Report No. 97001.

[15] B. L. Miller and D. E. Goldberg. Genetic Algorithms, Selection Schemes, and the Varying Effects of Noise. *Evolutionary Computation*, 4(2):113–131, 1997.

[16] L. M. Rattray. *Modelling the Dynamics of Genetic Algorithms Using Statistical Mechanics.* PhD thesis, University of Manchester, CS Department, Manchester, UK, 1996.

[17] M. Rattray and J. Shapiro. Noisy Fitness Evaluation in Genetic Algorithms and the Dynamics of Learning. In R. K. Belew and M. D. Vose, editors, *Foundations of Genetic Algorithms, 4.* Morgan Kaufmann, San Mateo, CA, 1997.

[18] D. H. Ackley. *A Connectionist Machine for Genetic Hillclimbing.* Kluwer Academic Publishers, Boston, 1987.

[19] E. T. Jaynes. Where Do We Stand on Maximum Entropy? In R.D. Levine and M. Tribus, editors, *The Maximum Entropy Formalism*, pages 15–118, 1979.

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

[21] I. Rechenberg. Evolutionsstrategien. In B. Schneider and U. Ranft, editors, *Simulationsmethoden in der Medizin und Biologie*, pages 83–114. Springer-Verlag, Berlin, 1978.

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

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

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

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

[26] T. Bäck and H.-P. Schwefel. An Overview of Evolutionary Algorithms for Parameter Optimization. *Evolutionary Computation*, 1(1):1–23, 1993.

[27] 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.

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

[29] H.-G. Beyer. An Alternative Explanation for the Manner in which Genetic Algorithms Operate. *BioSystems*, 41:1–15, 1997.

[30] 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.

[31] H.-G. Beyer. *Zur Analyse der Evolutionsstrategien.* Habilitationsschrift, University of Dortmund, 1996.

[32] H.-G. Beyer. Mutate Large, But Inherit Small! On the Analysis of Rescaled Mutations in $(\tilde{1}, \tilde{\lambda})$-ES with Noisy Fitness Data. In A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature, 5*, pages 109–118, Heidelberg, 1998. Springer.

[33] H.-G. Beyer. Towards a Theory of 'Evolution Strategies': Progress Rates and Quality Gain for $(1 \overset{+}{,} \lambda)$-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, 1994. Springer.

[34] M. G. Blumer. *The Mathematical Theory of Quantitative Genetics.* Clarendon Press, Oxford, 1980.