In Eshelman, L.J. (ed.), *Proceedings of the Sixth International Conference on Genetic Algorithms*: 57–64, 1995.
ERRATA: Sec.2.5.2, P.61: Use rather $c_r = 3/(n+3)$ and $\beta = 1/n$. Sec.2, P.58: See www.bionik.tu-berlin.de/user/niko

On the Adaptation of Arbitrary Normal Mutation Distributions in Evolution Strategies: The Generating Set Adaptation

Nikolaus Hansen*, Andreas Ostermeier & Andreas Gawelczyk
Fachgebiet für Bionik und Evolutionstechnik
Technische Universität Berlin
Ackerstr. 71–76
13355 Berlin, Germany

Abstract

A new adaptation scheme for adapting arbitrary normal mutation distributions in evolution strategies is introduced. It can adapt correct scaling and correlations between object parameters. Furthermore, it is independent of any rotation of the objective function and reliably adapts mutation distributions corresponding to hyperellipsoids with high axis ratio. In simulations, the *generating set adaptation* is compared to two other schemes which also can produce non axis-parallel mutation ellipsoids. It turns out to be the only adaptation scheme which is completely independent of the chosen coordinate system.

1 INTRODUCTION

In evolution strategies (ESs), a mutation is usually carried out by adding a $N(\vec{0}, \boldsymbol{A})$ distributed random vector[1]. The symmetric and positive semi-definite $n \times n$-matrix $\boldsymbol{A}$ represents the parameters of the mutation distribution. Assuming that the landscape of the objective function is unknown, in general $\boldsymbol{A}$ has to be adapted to get reasonable progress. The simplest way of adaptation is to confine $\boldsymbol{A}$ to $\delta^2\boldsymbol{I}$, where $\boldsymbol{I}$ denotes the identity matrix and δ denotes a global step size. Thus, δ is adapted. The mutation distribution remains isotropic, and the surfaces of isodensity are hyperspheres. Global step size adaptation was introduced by Rechenberg (1973) and Schwefel (1981) and is, in its mutative form, widely used in the ES community. Considering anisotropic distributions, we distinguish between two cases: The first, more specialized distribution has surfaces of equal density that are axis-parallel hyperellipsoids. $\boldsymbol{A}$ is a diagonal matrix, and there is still no correlation between coordinate axes. The corresponding adaptation mechanism for n variances was proposed by Schwefel (1981). The second, more general distribution can cause correlated mutations in the given coordinate system. $\boldsymbol{A}$ may not be diagonal anymore and has $n(n+1)/2$ free parameters. The surfaces of isodensity are arbitrarily orientated hyperellipsoids. A corresponding adaptation mechanism was proposed by Schwefel (1981) and analyzed by Rudolph (1992).

We will discuss three adaptation strategies, which can produce correlated mutations in the given coordinate system. They are described in Section 2. In Section 3, we introduce four objective functions, which can be used to reveal important aspects of coordinate system dependence and adaptation possibilities of the algorithms. Section 4 discusses corresponding simulation runs which reveal the different behaviors of the adaptation schemes. A conclusion is given in Section 5.

2 ADAPTATION SCHEMES

First, we will present the generating set adaptation (AI), a new approach to adaptation of arbitrary normal mutation distributions in (μ, λ)-ESs. Subsequently, a second adaptation scheme (AII) will be introduced, which cannot produce *arbitrary* mutation distributions, but seems to operate quite well on several objective functions which need correlated mutations for reasonable progress. We call these schemes derandomized, because the strategy parameters are not subject to direct mutations, but to the same (although transformed) stochastic variations as the object variables. Any *direct mutation*-selection scheme on *strategy* parameters is subject to considerable noise, because selection works on the adjustment of the object variables, while strategy parameters correspond

*E-mail: hansen@fb10.tu-berlin.de

[1] i.e. a normal distributed random vector with expectation zero and covariance matrix $\boldsymbol{A}$.

only in a loose (stochastic) way with object parameter changes.[2] The concept of derandomization was introduced by Ostermeier et al. (1994a). Both derandomized schemes will be formally described in Section 2.5, which, if the reader feels uncomfortable with the formalisms, can be skipped without breaking the continuity of the whole.

The third adaptation scheme (AIII) is due to Schwefel (1981), who introduced the idea of adapting *all* parameters of the normal distribution in ESs. This scheme is able to produce arbitrary mutation distributions, too.

None of these adaptation schemes causes additional evaluations of the objective function. However, generating the mutation vector in AI and AIII takes computational time in order of n^3. While in Schwefel's algorithm different kinds of recombinations are widely used, no sensible recombination operator is defined for the two derandomized schemes yet.

2.1 DERANDOMIZED ADAPTATION OF THE GENERATING SET (AI)

In the following, we try to *reveal the mechanism* of the generating set adaptation (AI) rather than being mathematically rigorous. To keep things clear, we consider the situation for one parent ($\mu = 1$).

Often, a mutation step is carried out by adding a normal distributed random vector $\boldsymbol{z}'$ on the object variable vector with

$$\boldsymbol{z}' := \delta \cdot (z_1 \, \boldsymbol{b}_1 + \ldots + z_n \, \boldsymbol{b}_n), \qquad (1)$$

where

n	number of object variables (dimension of the problem),
δ	global step size,
$z_i \sim N(0,1)$	for $i = 1, \ldots, n$, independent, $(0,1)$-normal distributed random numbers,
$\boldsymbol{b}_i := \boldsymbol{e}_i$	ith standard basis vector in $\mathbb{R}^n$.

$\boldsymbol{z}'$ is $N(\vec{0}, \delta^2 \boldsymbol{I})$ distributed. To get an anisotropic axis-parallel mutation ellipsoid, we can multiply each $z_i \, \boldsymbol{b}_i$ in equation (1) by a different individual step size σ_i. Furthermore, we can modify the distribution of $\boldsymbol{z}'$ by exchanging the $\boldsymbol{b}_i$ — thereby detaching $\boldsymbol{z}'$ from the given coordinate system. In this way, on the one hand *any* normal distribution can be produced. To see that, just consider sets of orthogonal $\boldsymbol{b}_i$. On the other hand the distribution is always normal, because (singular) normal distributions are summed up.

[2] E.g. a small mutation step in one coordinate is *not necessarily* produced by a corresponding small step size. Actually, mutation step length depends on step size *and random number realization* of the normal distributed mutation.

The Adaptation Process

Adaptation, independent of the coordinate system, will be achieved by successively exchanging the $\boldsymbol{b}_i$ in equation 1. Therefore, we are looking for a new vector to modify the mutation distribution in an appropriate way. We assume that the alteration of the best mutation distribution is slow concerning the generation sequence. Then the current "best" mutation step $\boldsymbol{z}_{\text{sel}}$ — i.e. the difference between object variable vectors of selected offspring and parent — yields most information obtainable about the best mutation distribution. Using $\boldsymbol{z}_{\text{sel}}$ in exchange for one $\boldsymbol{b}_i$ leads to the highest possible probability of producing mutation steps similar to $\boldsymbol{z}_{\text{sel}}$ in the future. Successively generating all $\boldsymbol{b}_i$ that way, mutation distribution depends on the landscape of the objective function, but is independent of the given coordinate system. To implement the adaptation mechanism, we take into account the following:

- We use not only n but — according to the number of free parameters to adapt — n^2 to $2n^2$ vectors $\boldsymbol{b}_i$. The vectors constitute a memory of selected mutation steps. The usefulness of such a memory had been suspected by Rudolph (1992). Of course, the necessary information can be collected in one generation as well as in the generation sequence by simply raising μ, i.e. the number of selected individuals per generation.

  In spite of using more than n vectors, all properties mentioned above are preserved. Especially, all produced distributions are normal, and all normal distributions with mean $\vec{0}$ can be produced.

- Only the oldest $\boldsymbol{b}_i$ will be exchanged. Thereby the most up-to-date information is always preserved.

- In addition, a separate global step size adaptation takes place. Thus, the size of the mutation step can be adjusted in a much shorter time scale than by adaptation of the generating system alone. The global step size adaptation is mutative, and the transmitted step size variations are damped by exponent β to suppress stochastic fluctuations. Section 2.5.1).

- The new $\boldsymbol{b}_i$ is calculated as (exponentially decreasing) weighted mean of all mutation steps of the individual's history, that is, all "best mutations" selected so far are accumulated. This accumulation yields non-local[3] selection information, whereby the sign[4] of the selected mutation steps

[3] i.e. non-local in time *and* (object parameter) space, where time refers to the generation sequence.

[4] i.e. whether the vector is orientated as it is, or whether it is orientated in the opposite direction.

influences the adaptation, because the weighted sum is not independent of the signs of the contributing vectors. The signs of the $\boldsymbol{b}_i$ themselves are insignificant, because they will be multiplied by $N(0,1)$ distributed random numbers.

The first point is essential, because the distribution tends to degenerate if there are too few vectors (without selection pressure, it tends to degenerate anyway). Damping of the transmitted *global* step size variation and accumulation of selection information are uncritical features and could be omitted. Storage capacity for the algorithm can be reduced from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$ by using a weighted sum of the covariance matrices of the random vectors $z\,\boldsymbol{z}_{\text{sel}}$, with $z \sim N(0,1)$ for all $\boldsymbol{z}_{\text{sel}}$, as covariance matrix of the adapted mutation distribution instead of storing the $\boldsymbol{b}_i$. The corresponding changes of the algorithm will not be discussed here.

2.2 DERANDOMIZED ADAPTATION OF n INDIVIDUAL STEP SIZES AND ONE DIRECTION (AII)

The derandomized adaptation of n individual step sizes (standard deviations) and one direction (AII) is an extension of the derandomized individual step size adaptation introduced by Ostermeier et al. (1994a, 1994b). In AII, the mutation distribution results from adding an uncorrelated normal distribution with axis-parallel hyperellipsoids as isodensity surfaces and an arbitrary one-dimensional (singular) normal distribution, namely a line mutation with expectation value zero. The first contribution is due to n individual step sizes, the second one to the adapted direction. Direction *adaptation* is done, basically speaking, by adding up the selected mutation steps in the generation sequence (accumulation). In other words, the line between great-great-grandparent and descendant serves as basis for direction adaptation. For individual step size adaptation, the main *functional* difference to a conventional mutative adaptation scheme is the damping of step size variations before transmitting them to the descendant (parameter β_{ind} in Section 2.5.2). Due to the axis-parallel contribution, the resulting distribution is not independent of the coordinate system. Furthermore, not every normal distribution can be produced.

2.3 ADAPTATION OF n STANDARD DEVIATIONS AND $n(n-1)/2$ ROTATION ANGLES (AIII)

Every n-dimensional normal distribution can be determined uniquely by n variances ${\sigma_i}^2$ and $n(n-1)/2$ rotation angles ω_j. The covariance matrix $\boldsymbol{A}$ can be determined by applying $n(n-1)/2$ elementary rotation matrices in a fixed order on the diagonal matrix $((\sigma_{ik})) := ((\delta_{ik}\,\sigma_i))$, where $\delta_{ik} \in \{0,1\}$ is the Kronecker symbol delta.[5] The matrix $((\sigma_{ik}))$ represents an axis-parallel mutation hyperellipsoid, which is rotated subsequently in every canonical plane. Schwefel (1981) proposed an adaptation scheme where the standard deviations σ_i and the rotation angles ω_j are mutated in the following way:

$$\begin{aligned} \sigma_i^{(g+1)} &= \sigma_i^{(g)}\; z\; z_i \\ \omega_j^{(g+1)} &= \left(\omega_j^{(g)} + z_j^{\omega} + \pi\right) \bmod 2\pi - \pi \end{aligned}$$

where

g generation,

$z \sim LN\left(0, \left(1/\sqrt{2n}\right)^2\right)$ logarithmic normal distributed, one realization for all σ_i of one generation g.

$z_i \sim LN\left(0, \left(1/\sqrt{2\sqrt{n}}\right)^2\right)$ for $i = 1, \ldots, n$,

$z_j^{\omega} \sim N\left(0, \left(\frac{5}{180}\,\pi\right)^2\right)$ for $j = 1, \ldots, n(n-1)/2$.

This adaptation scheme can produce any normal distribution with expectation $\vec{0}$.

2.4 GENERATING A DISTRIBUTION

To convey the idea how the different mechanisms produce a distribution, we give an example for the distribution $N\left(\binom{0}{0}, \binom{27\ 10}{10\ 15}\right)$ in $\mathbb{R}^2$. The three different methods to produce this distribution and its one-σ isodensity ellipsoid are shown in the following three figures. With respect to AII, $n = 2$ is a *very* special case, because *every* distribution can be generated in $\mathbb{R}^2$, whereas for $n \geq 3$ this is not true anymore.

Figure 1 shows a generating set consisting of the vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_4$. Adding up line mutations with respect to these vectors as in equation (1), where $\delta = 1$, leads to the shown distribution. Of course there are infinitely many vector sets which result in the same distribution.[6] In AI, the *adaptation process* simply replaces one of the vectors $\boldsymbol{b}_i$ with $\boldsymbol{z}_{\text{sel}}$ each generation (cf. Section 2.1). According to AII, Figure 2 shows one example for choosing two individual step sizes δ_1 and δ_2, each determining the length of an axis-parallel vector, and the direction vector $\boldsymbol{r}$. Again, adding up these vectors, each multiplied by a $N(0,1)$-distributed random number, results in the solid isodensity ellipsoid.

[5] If $\boldsymbol{B}$ is the result of the rotations, then $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^t$.

[6] The only vector set which generates the shown distribution *and* forms an orthogonal basis consists of the two thin lined vectors in Figure 3.

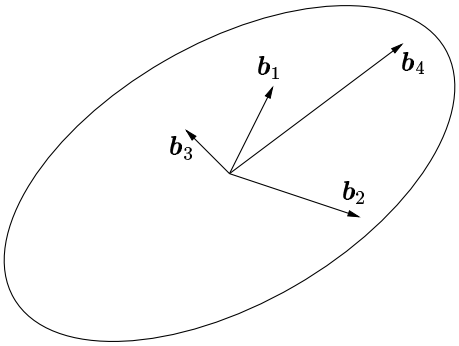

Figure 1: AI — A Generating Set and the Resulting Distribution

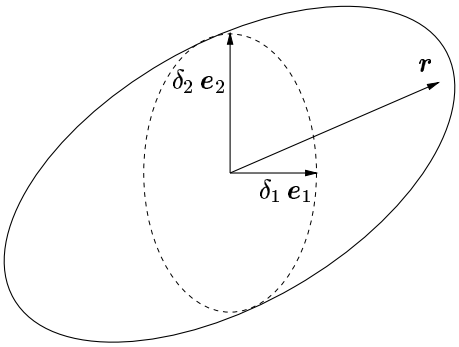

Figure 2: AII — The Distribution Produced by Individual Step Sizes And One Direction

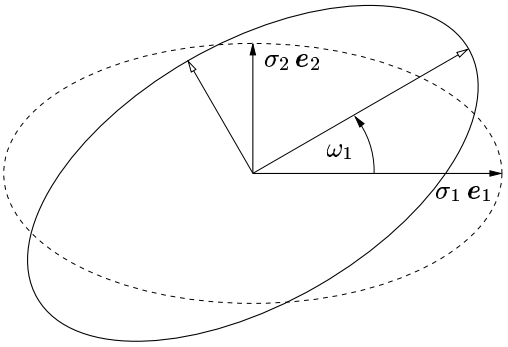

Figure 3: AIII — The Distribution Produced by Standard Deviations And Rotation Angle(s)

The dashed ellipsoid refers to the distribution resulting without direction vector. The *adaptation process* operates on δ_1, δ_2 and $\boldsymbol{r}$. In AIII, the mutation distribution is constructed by rotating an axis-parallel distribution. Correspondingly, the dashed ellipsoid in Figure 3 is rotated by $\omega_1 = (29.6/180)\pi$. The adapted parameters here are the standard deviations σ_1 and σ_2, which correspond to the vector lengths, and the rotation angle ω_1.

2.5 ALGORITHMS FOR AI AND AII

In this section, the algorithms AI and AII (cf. 2.1 and 2.2) are formally described for $\lambda > \mu \geq 1$. All random numbers used are independent, and index k denotes one realization for each $k = 1, \ldots, \lambda$. All vectors are column vectors and printed in bold faces. The following symbols are used repeatedly:

- n number of object parameters (dimension of the problem),
- $\boldsymbol{I}$ identity matrix,
- $\boldsymbol{x} = (x_1, \ldots, x_n)^t \in \mathbb{R}^n$ object variable vector,
- E_j index for j-th parent (Elder), $j = 1, \ldots, \mu$,
- N_k index for k-th offspring (Newer), $k = 1, \ldots, \lambda$,
- $\zeta_k = 1, \ldots, \mu$ with equal probability,
- $c_u = \sqrt{(2-c)/c}$ normalizes the variances of the left hand side, e.g. in equation (3), the factor c_u adjusts the variance of $\boldsymbol{b}_1^N$ to that of $\xi\, \boldsymbol{y}$.

For easier reading it is helpful to remove the non-essential accumulation by setting $c_u = c := 1$ and rewriting the equations (3), (4) and (5). Furthermore, if $\mu = 1$, the index ζ_k can be ignored.

2.5.1 Reproduction Scheme of AI

For $k = 1, \ldots, \lambda$ (i.e. for each offspring)

1. Realization of a normal distributed vector $\boldsymbol{y}$:

$$\begin{aligned} \boldsymbol{y}_k &= c_m\, \boldsymbol{B}^{E_{\zeta_k}}\, \boldsymbol{z}_k \\ &= c_m \sum_{j=1}^{m} (z_k)_j\, \boldsymbol{b}_j^{E_{\zeta_k}} \end{aligned} \qquad (2)$$

2. Mutation of object and strategy parameters:

$$\begin{aligned} \boldsymbol{x}^{N_k} &= \boldsymbol{x}^{E_{\zeta_k}} + \delta^{E_{\zeta_k}}\, \xi_k\, \boldsymbol{y}_k \\ \delta^{N_k} &= \delta^{E_{\zeta_k}}\, (\xi_k)^{\beta} \\ \boldsymbol{b}_1^{N_k} &= (1-c) \cdot \boldsymbol{b}_1^{E_{\zeta_k}} + c \cdot (c_u\, \xi_k\, \boldsymbol{y}_k) \\ \boldsymbol{b}_{i+1}^{N_k} &= \boldsymbol{b}_i^{E_{\zeta_k}} \quad \text{for } i = 1, \ldots, m-1 \end{aligned} \qquad (3)$$

where

- $\xi = 1.5,\ 1/1.5$ with equal probability ξ is the step size variation factor.
- $\boldsymbol{z} = (z_1, \ldots, z_m)^t \sim N(\vec{0},\ \boldsymbol{I})$ i.e. $z_i \sim N(0,1)$,
- δ global step size,
- $\boldsymbol{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_m) \in \mathbb{R}^n \times \mathbb{R}^m$ matrix of the generating set consisting of vectors $\boldsymbol{b}_i \in \mathbb{R}^n$. $\boldsymbol{B}$ transforms $\boldsymbol{z}$ from $\mathbb{R}^m$ into $\mathbb{R}^n$. Initialization: $\boldsymbol{b}_1^{E_j} = \vec{0}$ and $\boldsymbol{b}_2^{E_j}, \ldots, \boldsymbol{b}_m^{E_j} \sim N(0, (1/n)\, \boldsymbol{I})$, i.e. components of $\boldsymbol{b}_i$ $N(0, (1/\sqrt{n})^2)$ distributed.
- $\boldsymbol{b}_i \in \mathbb{R}^n$ vector of the generating set, see $\boldsymbol{B}$.

$m \in \{n^2, \ldots, 2n^2\}$ number of vectors of the generating set. The larger m, the more reliable, the smaller m, the faster is the adaptation. For simulations we have used $m = 1.5\,n^2$.

$c_m = (1/\sqrt{m})\,(1 + 1/m)$ adjusts the length of $\boldsymbol{y}$ in equation (2) so that $\|\boldsymbol{y}\| \approx \|\boldsymbol{b}_i\|$ holds and without selection, the length of all $\boldsymbol{b}_i$ remains about constant. The factor $(1 + 1/m)$ serves as approximation for small m.

$c = 1/\sqrt{n}$ found by simulations. c determines the accumulation time.

$\beta = 1/\sqrt{n}$ found by simulations. β determines the damping of the transmitted step size variation.

There are two stochastic sources in the reproduction scheme: $\boldsymbol{z}$ and ξ. The realization of both is used for object *and* strategy parameter mutation simultaneously.

2.5.2 Reproduction Scheme of AII

The description of adaptation scheme AII has been modified compared to Ostermeier et al. (1994b), but the adaptation process of the individual step sizes is virtually identical.

For $k = 1, \ldots, \lambda$ and $i = 1, \ldots, n$

1. Mutation of the object variables (componentwise)

$$x_i^{N_k} = x_i^{E\zeta_k} + \delta_i^{E\zeta_k}\, z_i^k + \delta_r{}^{E\zeta_k}\, z_r{}^k\, r_i^{E\zeta_k}$$

2. Adaptation of the individual step sizes

$$\boldsymbol{s}^{N_k} = (1-c)\cdot \boldsymbol{s}^{E\zeta_k} + c\cdot\left(c_u\,\boldsymbol{z}^k\right) \qquad (4)$$

$$\delta_i^{N_k} = \delta_i^{E\zeta_k} \cdot \overbrace{\exp\left\{\beta\left(\|\boldsymbol{s}^{N_k}\| - \widehat{\chi}_n\right)\right\}}^{\text{“global step size” adaptation}} \cdot \underbrace{\exp\left\{\beta_{\text{ind}}\left(\,|s_i^{N_k}| - \widehat{\chi}_1\right)\right\}}_{\text{individual step size adaptation}}$$

3. Direction adaptation

$$s_r{}^{N_k} = \max\left\{\begin{array}{l}(1-c)\cdot s_r{}^{E\zeta_k} + c\cdot\left(c_u\, z_r{}^k\right),\\ 0\end{array}\right. \qquad (5)$$

$$\boldsymbol{r}' = (1-c_r)\cdot\delta_r{}^{E\zeta_k}\,\boldsymbol{r}^{E\zeta_k} + c_r\cdot\left(\boldsymbol{x}^{N_k} - \boldsymbol{x}^{E\zeta_k}\right)$$

$$\boldsymbol{r}^{N_k} = \boldsymbol{r}'/\|\boldsymbol{r}'\|$$

$$\delta_r{}^{N_k} = \max\left\{\begin{array}{l}\delta_r{}^{E\zeta_k}\,\exp\left\{\beta_r\left(|s_r{}^{N_k}| - \widehat{\chi}_1\right)\right\},\\ \frac{1}{3}\left\|\boldsymbol{\delta}^{N_k}\right\|\end{array}\right.$$

where

$$\boldsymbol{z} = (z_1, \ldots, z_n)^t \sim N(\vec{0},\ \boldsymbol{I}) \quad \text{i.e. } z_i \sim N(0,1)$$

$z_r \sim N(0,1)$ random number for direction mutation,

$\boldsymbol{s} \in \mathbb{R}^n$ weighted sum of all realized random vectors $\boldsymbol{z}$ in the individual's history (accumulation). $\boldsymbol{s}$ is used for adaptation of individual step sizes. Initialization with $\vec{0}$.

s_r weighted sum for adaptation of step size δ_r (see also $\boldsymbol{s}$). Due to direction adaptation, values less than zero are unreasonable.

$\boldsymbol{\delta} = (\delta_1, \ldots, \delta_n)^t \in \mathbb{R}^n$ vector of individual step sizes,

δ_r step size for direction. If $\delta_r \ll \|\boldsymbol{\delta}\|$, adaptation would become a random walk due to a lack of selection relevance.

$\boldsymbol{r} \in \mathbb{R}^n$ direction vector, used to produce a line mutation,

$\widehat{\chi}_n = \sqrt{n}\,\left(1 - \frac{1}{4\,n} + \frac{1}{21\,n^2}\right)$ approximates the expectation of the χ_n-distribution,

$\widehat{\chi}_1 = \sqrt{2/\pi}$ expectation of the χ_1-distribution,

$c = \sqrt{1/n}$, $c_r = 3/n$, $\beta = 2/n$, $\beta_{\text{ind}} = 1/(4n)$, $\beta_r = \sqrt{1/(4n)}$ parameters, found by simulations. If $\beta = 0$, no adaptation of the corresponding step size(s) takes place.

2.5.3 Discussion of Parameters

$c, c_r \in\ ]0;1]$ determine the accumulation time. Roughly speaking, after $1/c$ generations about 2/3 of the original information has vanished. If $c = 1$, no accumulation takes place. Accumulation is essential for direction adaptation, because it is the only way to gather the needed selection information here. Therefore, difficult problems may require longer accumulation time which can be achieved by decreasing c_r.

$\beta, \beta_{\text{ind}}, \beta_r \in [0;1]$ are parameters for damping the step size variation transmissions. Increasing them leads to a faster, decreasing them leads to a more reliable adaptation of the corresponding strategy parameters. Therefore, if one δ_i drifts away, β_{ind} should be decreased.

3 OBJECTIVE FUNCTIONS

The three adaptation schemes have been tested with the following objective functions, where $n = 20$. First, as a suitable objective function to test scaling properties and coordinate system independence, we propose an arbitrarily orientated **hyperellipsoid** with a given ratio between longest and shortest axis (1000 here) and constant ratio between “adjacent” axes (1.44 for $n = 20$ here). We do not use $\sum_{i=1}^{20}((i)^{2.3}x_i)^2$, because

it looks too much like an ellipsoid with just one long axis: the ratio between the first two axes is 4.9 : 1, but 1.3 : 1 and 1.1 : 1 between the middle-most and the last both, respectively.

$$Q_1(\boldsymbol{x}) = \sum_{i=1}^{n} \left(1000^{\frac{i-1}{n-1}} \underbrace{\langle \boldsymbol{x}, \boldsymbol{e}_i \rangle}_{\| \; x_i}\right)^2$$

$$Q_2(\boldsymbol{x}) = \sum_{i=1}^{n} \left(1000^{\frac{i-1}{n-1}} \langle \boldsymbol{x}, \boldsymbol{o}_i \rangle\right)^2$$

where $\langle ., . \rangle$ denotes the canonical scalar product, and the vectors $\boldsymbol{o}_1, \ldots, \boldsymbol{o}_n \in \mathbb{R}^n$ form an orthonormal basis with random orientation. Q_1 is an axis-parallel, Q_2 a randomly orientated hyperellipsoid. We produce the ith basis vector $\boldsymbol{o}_i$ first as a vector with $N(0,1)$ distributed components, then subtract all its projections on the previously produced basis vectors and normalize the result. Using $\langle \boldsymbol{x}, \boldsymbol{o}_i \rangle$ instead of x_i can bring *any* objective function with domain in a subset of $\mathbb{R}^n$ into coordinate system independent orientation!

Second, we prefer a slightly different generalization of **Rosenbrock's function** than Schwefel (1981) suggested, where all $x_2, \ldots, x_n$ were interchangeable without changing the function at all. In our case, every x_i is correlated to its "neighbors" x_{i-1} and x_{i+1}:

$$Q_3(\boldsymbol{x}) = \sum_{i=1}^{n-1} \left(100 \left({x_i}^2 - x_{i+1}\right)^2 + (x_i - 1)^2\right)$$

$$Q_4(\boldsymbol{x}) = \sum_{i=2}^{n} \left(100 \left({x_i}^2 - x_{i-1}\right)^2 + (x_i - 1)^2\right)$$

The only difference between Q_3 and Q_4 is the reversed order of variables. During first stage of simulation, Rosenbrock's function requires continuous readaptation of the mutation distribution to achieve maximal progress.

4 SIMULATIONS AND DISCUSSION

The derandomized schemes AI and AII have been tested with a (1,10)-ES. Due to the adaptation mechanism, AIII needs larger population sizes and has been tested with a (15,100)-ES with intermediate recombination for object and strategy parameters, i.e. arithmetic mean of corresponding object and strategy variables of two parents. Other types of recombination (discrete, without) on object and strategy parameters do not improve the results. All simulations shown are typical out of at least five (AIII) or ten runs, and variances of these five or ten runs are clearly less than differences between shown simulations with different strategies. Single runs are chosen to show effects of the adaptation *process*, especially *changes* of progress over time.

Simulation results for the hyperellipsoid (Figure 4) and Rosenbrock's function (Figure 5) show that the generating set adaptation (AI) reliably adapts the mutation distribution to different objective function landscapes. Arbitrary, even rotated, hyperellipsoids are virtually transformed into the hypersphere: after the adaptation phase, the strategy realizes 80% of the progress rate that is possible with optimal mutation distribution. Corresponding with the theoretical considerations, AI is independent of rotations of the objective function (see Figure 4) and permutations of the coordinate axes (see Figure 5).

The disadvantage of AI is that the adaptation process takes a comparatively long time. On the hyperellipsoid, it takes about $4 \cdot 10^4$ function evaluations (descendants), as can be seen in Figure 4. Because of the number of free parameters of an arbitrary normal distribution, the adaptation time scales with n^2.

The adaptation process is faster when using AII. The mutation distribution is given by $2n$ free parameters, and the adaptation time scales with n. When adaptation is completed, the progress rate for the axis-parallel hyperellipsoid and for Rosenbrock's function are the same as with AI. Nevertheless, only special mutation distributions can be generated and, as expected, the algorithm is not independent of rotations of the objective function: Results on the arbitrarily orientated hyperellipsoid are significantly worse than on the axis-parallel one (see Figure 4).

Surprisingly, Schwefel's algorithm (AIII) depends drastically on rotations of the objective function and fails in adapting the arbitrarily orientated hyperellipsoid (see Figure 4). Even permutation of the coordinate axes affects the algorithm remarkably (see Figure 5). To verify this, we consider for $k = 1, \ldots, 10$ the objective functions

$$q_k = {x_k}^2 + \sum_{\substack{i=1 \\ i \neq k}}^{10} (100\, x_i)^2.$$

The landscapes of these hyperellipsoids are identical. The iso-fitness surface of each q_k looks like a (10-dimensional) cigar which is parallel to the k-th coordinate axis and has a ratio of 100:1 between length and diameter. For q_5, the progress rate of AIII turns out to be almost 10 times slower than for q_{10}. For

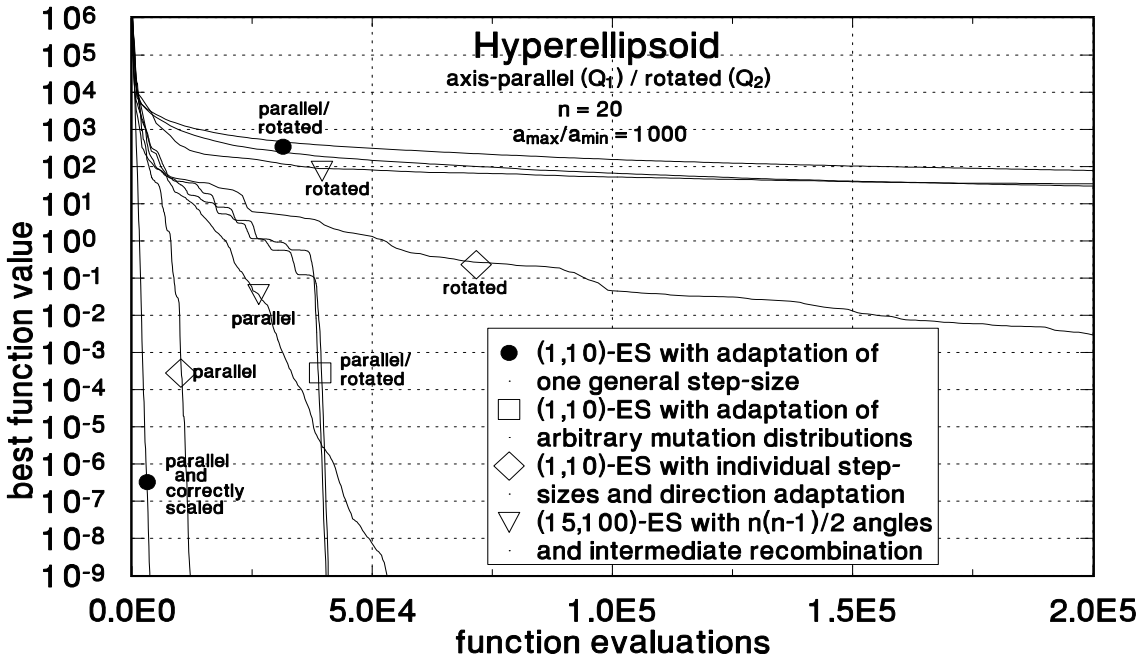

Figure 4: Simulation On the Hyperellipsoid. □ : AI, ◇ : AII, ▽ : AIII, ● : $(1, 10)$-ES with global step size adaptation. For every adaptation scheme one simulation with canonical (Q_1) and with randomly orientated orthonormal basis (Q_2), respectively, is shown. An additional run of the simple isotropic (1,10)-ES with global step size adaptation only, but correctly scaled individual step sizes (●) on Q_1 is shown for comparison. It illustrates nearly maximal progress for this type of strategy. Starting point of the simulation was $(1, \ldots, 1)^t$.

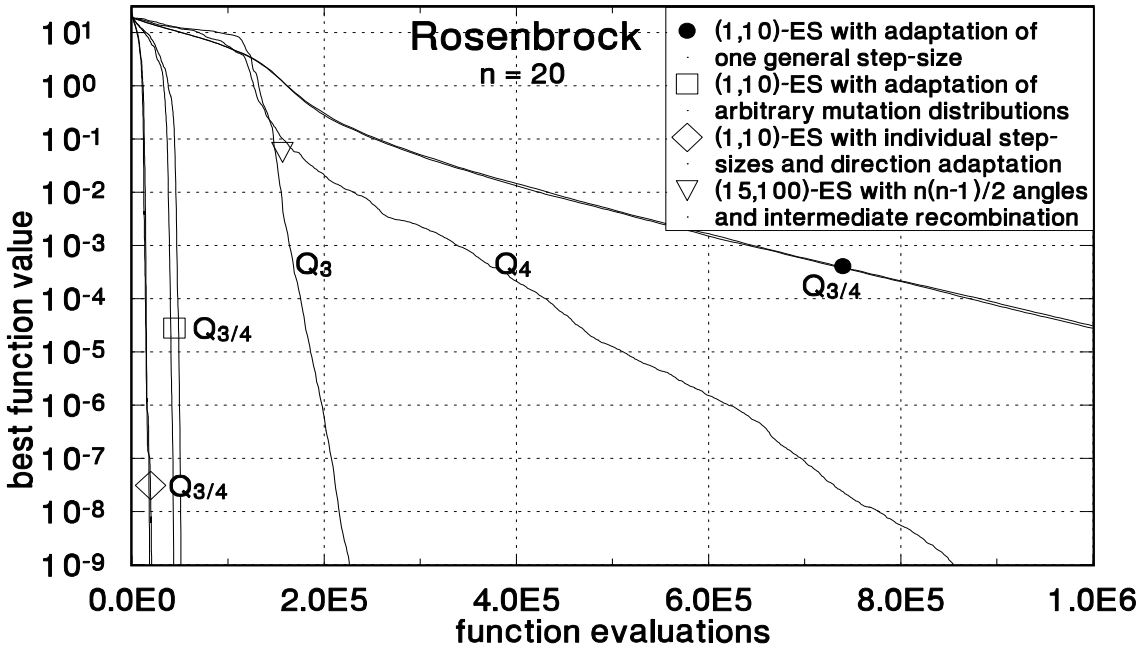

Figure 5: Simulation On Rosenbrock's Function. □ : AI, ◇ : AII, ▽ : AIII, ● : $(1, 10)$-ES with global step size adaptation. For every adaptation scheme one simulation with Q_3 and Q_4, respectively, is shown. For ◇ (AII) and ● both simulation runs are practically identical. For ▽ (AIII), in the final stage, progress rates differ by a factor seven. Starting point of the simulation was $\vec{0}$.

cigars in non axis-parallel positions,[7] the progress is even slower.

We interpret this behavior as caused by minor selection relevance of the angle positions, which for that reason are subject to serious stochastic fluctuations and perform almost random walks. Therefore, each angle ω_j should be equally distributed in the interval $[-\pi;\pi]$. What kind of mutation distribution does the rotation procedure generate if this interpretation is correct?

To answer this question, we take a standard basis vector $\boldsymbol{e}_i$, $i = 1,\ldots,n$ with equal probability, and transform it by elementary rotation matrices as described in Section 2.3, using in $[-\pi;\pi]$ equally distributed rotation angles. The angle α between the resulting vector and all coordinate axes, some diagonals and some random vectors is recorded. Relative frequencies of $\cos(\alpha) > 0.8$ are shown for $n = 20$ and $5 \cdot 10^6$ trials in Table 1. Obviously, an arbitrary unit vector is rotated into random or (nearly) diagonal position with considerably lower probability than into any (nearly) axis-parallel one. Furthermore, different axes have significantly different frequencies. This means that most of the axes of the distribution according to the diagonal matrix $((\sigma_{ik}))$ (cf. Section 2.3) are rotated in nearly axis-parallel positions again. Furthermore — depending on the order of the rotations — some coordinate axes are preferred to be parallel to the resulting mutation distribution which in consequence has comparatively high densities near some distinguished coordinate axes. This can explain all simulation results quite well. Consequently the assumption of random walks on the angles seems conclusive.

Table 1: Relative Frequencies of $\cos(\alpha) > 0.8$

Direction	Rel. Freq.
Axis with highest probability	$1.5 \cdot 10^{-2}$
Axis with lowest probability	$2.5 \cdot 10^{-4}$
Average of 20 random vectors	$8.3 \cdot 10^{-6}$
Average of 20 diagonals	$3.2 \cdot 10^{-7}$

5 CONCLUSIONS

This paper focuses on the adaptation of an arbitrary normal mutation distribution in evolution strategies and discusses two different schemes for this purpose: The generating set adaptation (AI), newly introduced here, proves to adapt all parameters of the normal mutation distribution reliably and independent of the coordinate system to an arbitrarily orientated hyperellipsoid even with high axis ratio. The adaptation of n variances and $n(n-1)/2$ rotation angles (AIII) turns out to be highly dependent on the chosen coordinate system, and cannot adapt arbitrary orientated hyperellipsoids. Because of its dependence on coordinate axis permutation, reproducibility depends on using identical order of objective variable definition and of rotation, respectively.

A general disadvantage of both adaptation schemes is, that the amount of selection information (i.e. the number of selected points in parameter space), which has to be gathered for a reliable adaptation, is of order n^2. Therefore in practical applications, it may be useful to restrict oneself to the adaptation of $2n$, n or just one (free) parameter(s), which could correspond to the adaptation of n variances and one direction (AII), n variances, or just the global step size, respectively. Especially AII should be taken into account, if the computational cost of the evaluation of a non-separable objective function is high, because it adapts its strategy parameters in a much faster time scale than AI and AIII and still works well on many of these functions.

[7] We have shown in Section 3, how to orientate a function arbitrarily.

Acknowledgements

This work was supported by the *BMBF* under grants 01 IB 404 A and 01 IN 107 A. We thank the anonymous reviewers for their helpful comments.

References

Ostermeier, A., Gawelczyk, A. & Hansen, N. (1994a). A Derandomized Approach to Self-Adaptation of Evolution Strategies. *Evolutionary Computation 2(4)*.

Ostermeier, A., Gawelczyk, A. & Hansen, N. (1994b). Step-size Adaptation Based on Non-local Use of Selection Information. In: Davidor, Y., Schwefel, H.-P. & Männer, R. (eds.), *Parallel Problem Solving from Nature – PPSN III, Proceedings*: pp. 189–198. Berlin: Springer.

Rechenberg, I. (1973). Evolutionsstrategie, Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. In Rechenberg, I. (1994), *Evolutionsstrategie '94*, Stuttgart: frommann-holzboog.

Rudolph, G. (1992). On Correlated Mutations in Evolution Strategies. In: Männer, R. & Manderick, B. (eds.), *Parallel Problem Solving from Nature, 2, Proceedings*: pp. 105–114. Amsterdam: North-Holland.

Schwefel, H.-P. (1981). *Numerical optimization of computer models.* Chichester: Wiley.