Invariance, Self-Adaptation and Correlated Mutations in Evolution Strategies

Nikolaus Hansen

Technische Universität Berlin
Fachgebiet für Bionik und Evolutionstechnik, Sekr. ACK 1
Ackerstr. 71–76, 13355 Berlin, Germany
E-mail: hansen@bionik.tu-berlin.de
URL: www.bionik.tu-berlin.de/user/niko/

Abstract. A conceptual objective behind the self-adaptation of the mutation distribution is to achieve invariance against certain transformations of the search space. In this paper, a priori invariances of a simple evolution strategy and invariances, which can be introduced by self-adaptation, are identified. In principle, correlated mutations can achieve invariance against any linear transformation of the search space. Correlated mutations, as typically implemented, are investigated with respect to both a priori and new invariances. Simulations reveal that neither all a priori invariances are retained, nor the invariance against linear transformation is achieved.

1 Introduction

The evolution strategy (ES) [9, 13] addresses the search problem of minimizing a nonlinear objective function $f : \mathcal{S} \subseteq \mathbb{R}^n \to \mathbb{R}, \boldsymbol{x} \mapsto f(\boldsymbol{x})$. Search steps are taken by recombination of already evaluated search points and mutation. The mutation is usually carried out by adding a realization of a normally distributed random vector. A dynamic control of certain parameters of the normal mutation distribution is of major importance and is a common feature in evolution strategies. This is often called adaptation or self-adaptation.

A main objective of the adaptation of parameters of the mutation distribution can be interpreted as to achieve invariance against certain transformations of the search space. This is exemplified in the following.

Let the mutation step for the object parameter vector, $\boldsymbol{x} \in \mathbb{R}^n$, at generation $g = 0, 1, 2, \ldots$ be

$$\boldsymbol{x}^{(g+1)} = \boldsymbol{x}^{(g)} + \sigma^{(g+1)} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \quad , \tag{1}$$

where $\sigma \in \mathbb{R}^+$ denotes the step size and $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ is a component wise independently $(0, 1)$-normally distributed random vector ($\boldsymbol{I}$ denotes the unity matrix as covariance matrix). Consider an adaptation of step size σ minimizing $f : \boldsymbol{x} \mapsto q(\boldsymbol{x})$ with $q : \mathbb{R}^n \to \mathbb{R}$ and initial values $\boldsymbol{x}^{(0)} = \boldsymbol{a}$ and $\sigma^{(0)} = 1$. With a sensible step size adaptation the following should be achieved.

- The strategy shows *exactly* the same behavior on $f : \boldsymbol{x} \mapsto q(10\boldsymbol{x})$ with initial values $\boldsymbol{x}^{(0)} = 0.1\boldsymbol{a}$ and $\sigma^{(0)} = 0.1$. This means (in general) invariance against a scalar multiplication of the object parameter vector and will be discussed below.
- Assume $\sigma^{(0)} = 10$ to be optimal. Then, with initial value $\sigma^{(0)} = 1$, the step size increases within the next generations to become (nearly) optimal. This is a quite obvious objective of step size adaptation.

Generalizing this perspective, two basic demands on the adaptation of any parameters of the mutation distribution can be stated.

1. Introducing (new) invariances.
2. Performing, after an adaptation phase, comparable to the best fixed parameter setting in this situation on (all) relevant objective functions [3, 15].

Together with a sufficiently fast adaptation rate these properties are of major importance for a sensible adaptation mechanism.

The invariance which can be achieved by the adaptation of certain strategy parameters is usually easy to identify (see below). The fundamental starting hypothesis of this paper is, that this invariance is a necessary condition for a successful and reliable adaptation. Starting from the best strategy parameter setting, with adaptation turned on, this invariance ensures a predictable behavior of the strategy (compare the example above assuming $\sigma^{(0)}$ to be optimal). Far from being a sufficient demand, this is probably the strongest non-empirical result one can hope for when evaluating an adaptation procedure. The relevance of this hypothesis can be reviewed in Fig. 2 (Sect. 4.2).

Concerning the parameters of the normal mutation distribution, which are dynamically controlled, in the most general case the complete covariance matrix is adapted. It is not too difficult to prove, that the choice of a certain covariance matrix can be identified with the choice of a linear transformation of the object parameter vector and vice versa [3]: Assume two linear transformations, T_A and T_B, and $\boldsymbol{y} = T_\mathrm{A}\boldsymbol{x}_\mathrm{A} = T_\mathrm{B}\boldsymbol{x}_\mathrm{B}$. The "genotypes" $\boldsymbol{x}_\mathrm{A}$ and $\boldsymbol{x}_\mathrm{B}$ can be interpreted as different codings for the same "phenotype", $\boldsymbol{y}$, which is used to evaluate the fitness. The effect of different codes becomes evident, when mutation is applied:

$$\begin{aligned}
\boldsymbol{y}_\mathrm{new} = T_\mathrm{B}\left(\boldsymbol{x}_\mathrm{B} + \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})\right) &\overset{T_\mathrm{B}\text{ linear}}{=} T_\mathrm{B}\boldsymbol{x}_\mathrm{B} + T_\mathrm{B}\mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \\
&\overset{T_\mathrm{A}\text{ bijective}}{=} T_\mathrm{A}\boldsymbol{x}_\mathrm{A} + T_\mathrm{A}{T_\mathrm{A}}^{-1}T_\mathrm{B}\mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \\
&\overset{T_\mathrm{A}\text{ linear}}{=} T_\mathrm{A}\left(\boldsymbol{x}_\mathrm{A} + {T_\mathrm{A}}^{-1}T_\mathrm{B}\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})\right)
\end{aligned}$$

Using a new coding, $T_\mathrm{A}\boldsymbol{x}_\mathrm{A}$ instead of $T_\mathrm{B}\boldsymbol{x}_\mathrm{B}$, is equivalent with introducing a certain linear transformation, ${T_\mathrm{A}}^{-1}T_\mathrm{B}$, for $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, which is equivalent with choosing a different covariance matrix, ${T_\mathrm{A}}^{-1}T_\mathrm{B}\left({T_\mathrm{A}}^{-1}T_\mathrm{B}\right)^\mathrm{t}$ instead of $\boldsymbol{I}$, for the mutation distribution.

While the choice of the initial strategy parameters (the initial covariance matrix) clearly introduces a dependency on the given coding, this dependency may vanish for $g \to \infty$, if the algorithm adapts the covariance matrix *and* is

invariant against linear transformations of the object parameters (and strategy parameters accordingly). Therefore the adaptation of the complete covariance matrix should strive for becoming independent from any linear transformation of the object parameters.

The importance of the interaction between problem encoding (e.g. linear transformation of the object parameters) and operators (e.g. the mutation operator) and invariance against certain transformations is in general well recognized [8, 11, 14, 15]. Why invariance is important from a practical point of view is also discussed in the next section.

The objective of this paper is to identify invariance properties of the ES and investigate correlated mutations, as typically applied, with respect to these invariance properties.

2 Invariance

Invariance properties of a search algorithm with respect to changes of the objective function are extremely attractive. They raise the probability to get similar (or even identical) results on similar, not yet evaluated objective functions and increase the predictability of the strategy behavior. *Any* evaluation of search algorithms is based on the implicit assumption of getting similar results on not yet evaluated functions.

A simple $(1, \lambda)$- or $(1 + \lambda)$-evolution strategy has the following invariance properties, where $q : \mathbb{R}^n \to \mathbb{R}$ is chosen arbitrarily [3]:

Translation: On $f : \boldsymbol{x} \mapsto q(\boldsymbol{x} - \boldsymbol{a})$ invariance against $\boldsymbol{a} \in \mathbb{R}^n$, if the initial object parameter $\boldsymbol{x}^{(0)}$ is chosen properly.

Rotation: On $f : \boldsymbol{x} \mapsto q(\boldsymbol{U}\boldsymbol{x})$ invariance against the orthogonal matrix $\boldsymbol{U}$, if the initial object parameter $\boldsymbol{x}^{(0)}$ is chosen properly. This means invariance against the chosen (orthonormal) coordinate system and includes invariance against rotation and reflection of the search space.

Order-preserving transformation: On $f : \boldsymbol{x} \mapsto g(q(\boldsymbol{x}))$ invariance against the monotonically increasing, i.e. order-preserving, function $g : \mathbb{R} \to \mathbb{R}$.

All three invariances can be achieved simultaneously: On $f : \boldsymbol{x} \mapsto g(q(\boldsymbol{U}\boldsymbol{x} - \boldsymbol{a}))$ the simple ES is invariant against g, $\boldsymbol{U}$, and $\boldsymbol{a}$ as described above. Invariance can be lost if more complex operators are introduces in the ES. Two common cases immediately come in mind where invariance against rotation is lost, because the introduced operator depends on the given coordinate axes:

- Individual step sizes.
- Discrete recombination on object parameters (parent number μ greater than one).

In general, the advantage of introducing such operators must be weighted carefully against the disadvantage of losing certain invariance properties.

Strategy parameter control can yield additional invariances. When individual step sizes are introduced, on $f : \boldsymbol{x} \mapsto q(\boldsymbol{D}\boldsymbol{x})$ invariance against the full rank

diagonal matrix $\boldsymbol{D}$ can be achieved (in exchange with rotation invariance). In other words invariance against a scaling with respect to the given coordinate system.

It is worth noting that the formulation of the adaptation mechanism plays an important role. Only a suitable formulation can achieve new invariances. Referring to (1), consider $f : \boldsymbol{x} \mapsto q(\boldsymbol{x})$ with initial values $\sigma^{(0)} = 1$ and $\boldsymbol{x}^{(0)} = \boldsymbol{a}$. If step size control is applied and σ is varied by adding/subtracting a constant value, say 0.05, the strategy cannot perform identical on $f : \boldsymbol{x} \mapsto q(10\boldsymbol{x})$ with $\sigma^{(0)} = 0.1$ and $\boldsymbol{x}^{(0)} = 0.1\boldsymbol{a}$, because $q(\boldsymbol{x}^{(1)})$, where $\boldsymbol{x}^{(1)} = \boldsymbol{a}+(1+0.05)\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, is different from $q(10\boldsymbol{x}^{(1)})$, where $\boldsymbol{x}^{(1)} = 0.1\boldsymbol{a}+(0.1+0.05)\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. In the latter the change of σ appears to be much larger. If σ is varied by multiplying/dividing by a constant value, say 1.05, the strategy *can* perform identical on $f : \boldsymbol{x} \mapsto q(10\boldsymbol{x})$. In this case $q(\boldsymbol{x}^{(1)})$, where $\boldsymbol{x}^{(1)} = \boldsymbol{a}+(1\cdot 1.05)\mathcal{N}_k(\boldsymbol{0}, \boldsymbol{I})$, is identical with $q(10\boldsymbol{x}^{(1)})$, where $\boldsymbol{x}^{(1)} = 0.1\boldsymbol{a} + (0.1 \cdot 1.05)\mathcal{N}_k(\boldsymbol{0}, \boldsymbol{I})$.

In contrast to an individual step size adaptation, in certain cases new invariances can be introduced without giving up existing ones. The following two *additional* invariances can be achieved by strategy parameter control, if initial object and strategy parameters are chosen accordingly:

Scalar multiplication: On $f : \boldsymbol{x} \mapsto q(c \cdot \boldsymbol{x})$ invariance against the scalar $c \neq 0$, if the initial step size is chosen properly. The typical mutative step size control in evolution strategies achieves the invariance against scalar multiplication.

Linear transformation: On $f : \boldsymbol{x} \mapsto q(\boldsymbol{A} \cdot \boldsymbol{x})$ invariance against any full rank $n \times n$-matrix $\boldsymbol{A}$, if the initial covariance matrix of the mutation distribution is chosen properly. When the complete covariance matrix of the mutation distribution is adapted, invariance against linear transformation can be achieved.

The question arises, whether the typically applied mutative adaptation scheme for the complete covariance matrix [12, 13], usually referred to as correlated mutations, actually achieves invariance against linear transformation (and retains invariances of a simple ES). On the one side this scheme, as formalized below, is invariant against translation, order-preserving transformation and scalar multiplication. On the other side former investigations indicate that the scheme has lost invariance against rotation [6, 7]. Nevertheless it is still unclear, whether invariance against linear transformation is achieved if object *and* strategy parameters are transformed accordingly. In Sect. 4 the invariance properties against rotation and linear transformation are investigated in detail.

3 Correlated Mutations

Correlated mutations were proposed in [12] and have become a standard algorithm [1, 2] which seems especially promising for difficult problem instances. This algorithm is denoted with CORR-ES. For this paper a $(15/2_{\mathrm{I}}, 100)$-CORR-ES

is implemented, which denotes an evolution strategy with 15 parents, 100 offspring and with intermediate two parent recombination.[1] In the CORR-ES the normal distribution with zero mean is parameterized through n step sizes, σ_i, and $n(n-1)/2$ inclination angles, α_j. At generation g for each offspring two parents are selected. The component wise arithmetic mean of step sizes, angles and object parameter vector of the two parents ($\sigma^{(g)}_{i=1,\ldots,n}$, $\alpha^{(g)}_{j=1,\ldots,n(n-1)/2}$ and $\boldsymbol{x}^{(g)}$) are starting-points for the mutation. For step sizes and angles the mutation reads component wise

$$\sigma_i^{(g+1)} = \sigma_i^{(g)} \cdot \exp\left(\mathcal{N}\left(0, \frac{1}{2n}\right) + \mathcal{N}_i\left(0, \frac{1}{2\sqrt{n}}\right)\right) \tag{2}$$

$$\alpha_j^{(g+1)} = \left(\alpha_j^{(g)} + \mathcal{N}_j\left(0, \left(\frac{5}{180}\pi\right)^2\right) + \pi\right) \bmod 2\pi - \pi \tag{3}$$

The random number $\mathcal{N}(0, 1/(2n))$ in (2), denoting a normal distribution with zero mean and variance $1/(2n)$, is only drawn once for all $i = 1, \ldots, n$. The modulo operation ensures the angles to be in the interval $-\pi \leq \alpha_j^{(g+1)} < \pi$, which is, to my experience, only of minor relevance. The object parameter mutation reads

$$\boldsymbol{x}^{(g+1)} = \boldsymbol{x}^{(g)} + R\left(\alpha_1^{(g+1)}, \ldots, \alpha_{n(n-1)/2}^{(g+1)}\right) \cdot \begin{pmatrix} \sigma_1^{(g+1)} & & \\ & \ddots & \\ & & \sigma_n^{(g+1)} \end{pmatrix} \cdot \mathcal{N}(\mathbf{0}, \boldsymbol{I}) \tag{4}$$

The $(\mathbf{0}, \boldsymbol{I})$-normally distributed random vector is transformed with a diagonal matrix determined by $\sigma_1^{(g+1)}, \ldots, \sigma_n^{(g+1)}$. The result is rotated in all two-dimensional subspaces spanned by canonical unit vectors, denoted with $R(.)$. The resulting distribution is a normal distribution and any normal distribution with zero mean can be generated this way [10]. Therefore the algorithm implements an adaptation of the complete covariance matrix. Note, that for a replicable definition of the algorithm the order of the chosen subspaces has to be defined as well. (The order can have a considerable impact on the performance result as can be concluded from the results shown below.) In this paper the coordinate numbers are chosen in the ordering $(1,2), (1,3), \ldots, (1,n), (2,3), (2,4), \ldots, (2,n), \ldots (n-1,n)$. Initial values are chosen $\boldsymbol{x}^{(0)} = (1, \ldots, 1)^{\mathrm{t}}$, $\sigma_i^{(0)} = 1$ for $i = 1, \ldots, n$ and $\alpha_j^{(0)} = 0$ for $j = 1, \ldots, n(n-1)/2$, if not stated otherwise.

4 Invariance and Correlated Mutations

The rotation procedure of CORR-ES operates with respect to the given coordinate system. This leads to the assumption that the algorithm is not independent

[1] Rotation invariance and invariance against linear transformation can only be achieved, if discrete recombination on object parameters is avoided.

FOR $i = 1$ TO n
1. Draw components of $\boldsymbol{u}_i$ independently $(0, 1)$-normally distributed
2. $\boldsymbol{u}_i := \boldsymbol{u}_i - \sum_{j=1}^{i-1} \langle \boldsymbol{u}_i, \boldsymbol{u}_j \rangle \boldsymbol{u}_j$ ($\langle ., . \rangle$ denotes the canonical scalar product)
3. $\boldsymbol{u}_i := \boldsymbol{u}_i / \|\boldsymbol{u}_i\|$

ROF

Fig. 1. Algorithm to generate a random orthogonal matrix $\boldsymbol{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n]$ [6].

of the given coordinate system and therefore not invariant against rotation and linear transformation. Therefore, in this section, invariance is evaluated quantitatively by experiments on the following test functions.

4.1 Test Functions

All used objective functions are linear transformations of f_{sphere}, that is, they can be written in the form $f : \boldsymbol{x} \mapsto f_{\text{sphere}}(\boldsymbol{A} \cdot \boldsymbol{x})$, where $\boldsymbol{A}$ is a certain full rank $n \times n$-matrix. **Table 1** gives the used functions.

Table 1. Test Functions to be minimized, $k = 1, \ldots, n$

$f_{\text{sphere}}(\boldsymbol{x}) := \sum_{i=1}^{n} x_i^2$
$f_{\text{cigar}(k)}(\boldsymbol{x}) := \sum_{i=1}^{n} (a_i x_i)^2$, where $a_i = \begin{cases} 1 & \text{if } i = k \\ 100 & \text{otherwise} \end{cases}$
$f_{\text{tablet}(k)}(\boldsymbol{x}) := \sum_{i=1}^{n} (a_i x_i)^2$, where $a_i = \begin{cases} 100 & \text{if } i = k \\ 1 & \text{otherwise} \end{cases}$
$f_{\text{elli}(k)}(\boldsymbol{x}) := \sum_{i=1}^{n} (a_i x_i)^2$, where $a_i = \begin{cases} 100^{\frac{i}{n-1}} & \text{if } i < k \\ 1 & \text{if } i = k \\ 100^{\frac{i-1}{n-1}} & \text{if } i > k \end{cases}$

For $k = 1, \ldots, n$ the differences between the functions, e.g., $f_{\text{cigar}(k)}$ are due to permutations of the coordinate axes. These permutations can be interpreted as orthogonal linear transformations of the object parameter space, which leave $\sigma^{(0)}_{i=1,\ldots,n} = 1$ and $\boldsymbol{x}^{(0)} = (1, \ldots, 1)^{\mathrm{t}}$, as given in Sect. 3, unchanged. For $n = 5$ the coefficients a_i of $f_{\text{elli}(k)}$, where $k = 1, \ldots, 5$, are permutations of the set $\{1, 100^{\frac{1}{4}}, 100^{\frac{1}{2}}, 100^{\frac{3}{4}}, 100\}$.

Additionally f_{cigarU}, and correspondingly f_{tabletU} and f_{elliU}, is defined as

$$f_{\text{cigarU}}(\boldsymbol{x}) := f_{\text{cigar}(1)}(\boldsymbol{U}\boldsymbol{x})$$

with $\boldsymbol{x}^{(0)} = \boldsymbol{U}^{-1}(1, \ldots, 1)^{\mathrm{t}}$, where $\boldsymbol{U}$ is a random orthogonal (i.e. unitary) matrix chosen anew for each run (see **Fig. 1**). This implements an arbitrary rotation (and reflection) of the search space and basically leaves the topology of the function unchanged. Apart from f_{cigarU}, f_{tabletU} and f_{elliU} all functions are separable and can be solved without correlated mutations.

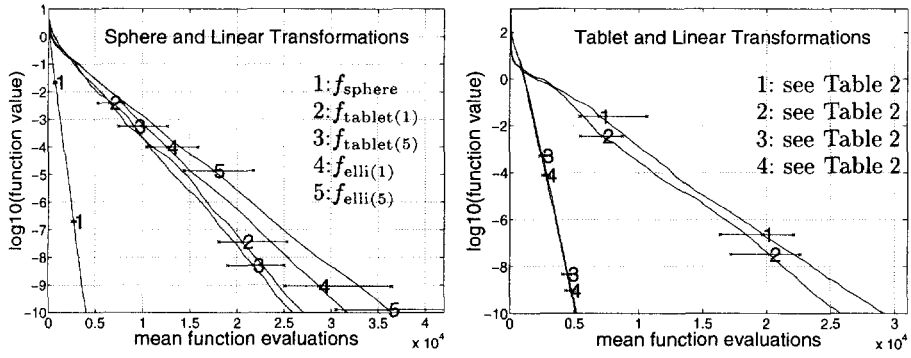

Fig. 2. Testing invariance against linear transformation. Shown are the mean number of function evaluations (abscissa) to reach a certain function value (ordinate) from 20 runs. The numbers in the right figure correspond to the numbers in Table 2. A strategy which is invariant against linear transformations must show identical graphs in each figure, because the initial values for object and strategy parameters are transformed according to f_{sphere} in the left and according to f_{tablet} in the right. Left and right error bars depict $\sqrt{\text{mean}_{x_i<\overline{x}}(x_i-\overline{x})^2}$ and $\sqrt{\text{mean}_{x_i>\overline{x}}(x_i-\overline{x})^2}$.

Table 2. Functions and initial values for Fig. 2, right

1: $f_{\text{tablet}(1)}$	$\sigma^{(0)}_{i=1,\ldots,5} = (1,1,1,1,1)$	$\boldsymbol{x}^{(0)} = (1,1,1,1,1)^{\text{t}}$
2: $f_{\text{tablet}(5)}$	$\sigma^{(0)}_{i=1,\ldots,5} = (1,1,1,1,1)$	$\boldsymbol{x}^{(0)} = (1,1,1,1,1)^{\text{t}}$
3: f_{sphere}	$\sigma^{(0)}_{i=1,\ldots,5} = (100,1,1,1,1)$	$\boldsymbol{x}^{(0)} = (100,1,1,1,1)^{\text{t}}$
4: f_{sphere}	$\sigma^{(0)}_{i=1,\ldots,5} = (1,1,1,1,100)$	$\boldsymbol{x}^{(0)} = (1,1,1,1,100)^{\text{t}}$

All simulations are carried out with problem dimension $n = 5$. Larger dimensions yield more pronounced effects but have a too large CPU time consumption to yield all needed results in the given time. Because invariance is investigated, there seems no particular need to do simulations in a wide range of problem dimensions.

4.2 Simulations

All simulations were carried out with the $(15/2_{\text{I}}, 100)$-CORR-ES and $n = 5$. In **Fig. 2**, left, simulations on f_{sphere} are shown, compared to simulations on f_{tablet} and f_{elli}, where object and strategy parameters are linearly transformed accordingly to f_{sphere}. For $f(\boldsymbol{x}) = \sum_i (a_i x_i)^2$ this means $\boldsymbol{x}^{(0)} = (a_1^{-1}, \ldots, a_n^{-1})^{\text{t}}$ and $\sigma_i^{(0)} = a_i^{-1}$ for $i = 1, \ldots, n$ (and $\alpha_j^{(0)} = 0$ for $j = 1, \ldots, n(n-1)/2$). This initial mutation distribution is optimal with respect to the objective function topology. It transforms the function into the sphere model and, if held constant, yields progress rates like on f_{sphere}. Even though the initial distribution *is* chosen optimal, performance on f_{tablet} and f_{elli} is worse by a factor of six to nine compared to f_{sphere}. The CORR-ES is not able to keep the deviation from the

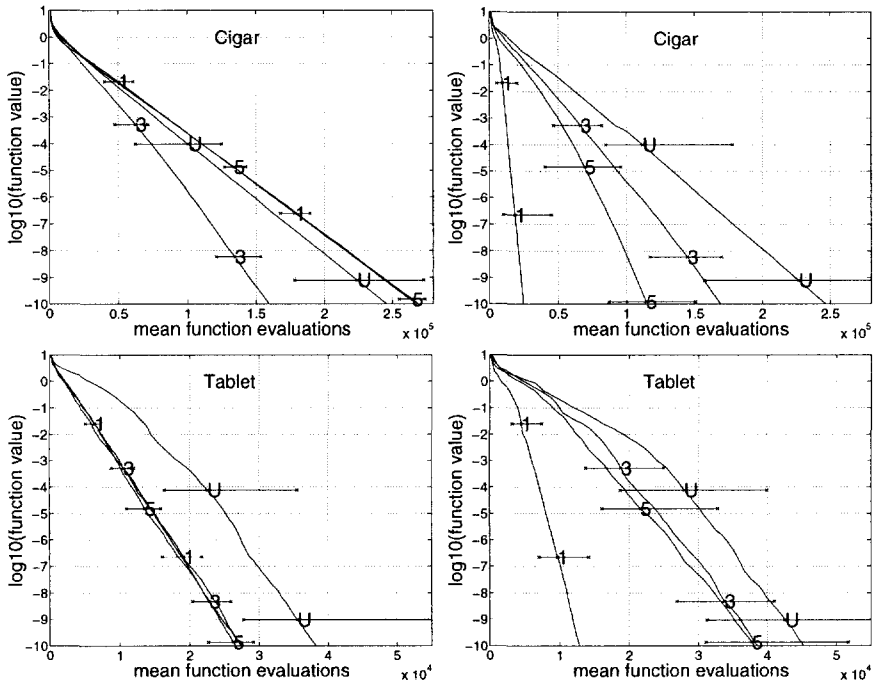

Fig. 3. Testing invariance against rotation. Shown are the mean number of function evaluations (abscissa) to reach a certain function value (ordinate) from 20 runs. The numbers depicting the graphs correspond to the number k in Table 1. That means, in the upper left runs on f_{cigarU}, $f_{\text{cigar}(1)}$, $f_{\text{cigar}(3)}$ and $f_{\text{cigar}(5)}$ are shown. In the left column initial angles are zero, in the right column they are uniformly randomly chosen in $[-\pi, \pi]$ alike for all initial parents. *For all graphs in each figure the topology of the function is identical.* Left and right error bars depict $\sqrt{\text{mean}_{x_i<\overline{x}}(x_i-\overline{x})^2}$ and $\sqrt{\text{mean}_{x_i>\overline{x}}(x_i-\overline{x})^2}$.

optimal initial distribution sufficiently small and is not invariant against linear transformations.

In the right of Fig. 2 simulations on $f_{\text{tablet}(1)}$ are shown together with simulations on $f_{\text{tablet}(5)}$ and f_{sphere}, where $\boldsymbol{x}^{(0)}$ and $\sigma_i^{(0)}$ are transformed accordingly to $f_{\text{tablet}(1)}$ or $f_{\text{tablet}(1)}$, as shown in Table 2. Even though the initial distribution is wrong, performance on f_{sphere} is only slightly effected and five times better than on $f_{\text{tablet}(1)}$. The CORR-ES clearly "remembers" the given coordinate system with the given scaling, which is quite advantageous (only) on f_{sphere}.

Invariance against rotation (precisely against orthogonal transformation) is exploited in **Fig. 3**. For example, the difference between $f_{\text{cigar}(1)}$ and $f_{\text{cigar}(3)}$ is an exchange of the coordinate axes one and three, which is a special orthogonal transformation. Simulation results on f_{elli} are similar to those on f_{tablet} and omitted due to limited space. In the CORR-ES the rotation procedure is the only part which can interact with an exchange of coordinate axes in the objec-

tive function (assuming in particular $\sigma_1^{(0)} = \ldots = \sigma_n^{(0)}$, see Sect. 4.1). Therefore, exchanging coordinate axes is equivalent with exchanging the corresponding coordinate numbers in the ordering of subspaces used in the rotation procedure.

In the left of Fig. 3 initial angles are zero. Only on $f_{\text{cigar}(3)}$ the exchange of coordinate axes has a remarkable impact on the performance. Compared to the axis-parallel versions, $f_{\text{tablet}(1;3;5)}$, on the arbitrarily oriented version, $f_{\text{tablet}\,\mathbf{U}}$, the performance is slightly worse with considerably larger variance. Recall that each of these runs is performed on a different function because the orthonormal basis $\boldsymbol{U}$ is chosen anew for each run.

If the initial angles are chosen uniformly randomly in $[-\pi, \pi]$ alike for all initial parents, the coordinate system dependency becomes more pronounced (right column in Fig. 3). On both functions an exchange of coordinate axes in the problem formulation has a remarkable impact on the performance (up to a factor of seven here). In particular the performance for $k = 1$ is remarkably improved. Recall that these objective functions are completely separable. If the coordinate system is chosen arbitrarily (like in $f_{\text{cigar}\,\mathbf{U}}$), performance is worst.

The original intention in the CORR-ES to get, after a transition phase, progress rates like on f_{sphere} [13, p. 243] cannot be met in any of these simulations. The observed progress rates are worse by a factor between 2.5 ($f_{\text{tablet}(1)}$, Fig. 3 right) and 70 ($f_{\text{cigar}(1)}$, Fig. 3 left) compared to f_{sphere}.

Interpreting previous results [6, 7] and further simulations not shown here, the author suspects the performance differences to become (much) larger when n is increased and/or the ratio between μ and n^2 becomes considerably smaller than $15/5^2$ (which equals μ/n^2 in the shown results).

5 Conclusion

Invariance is an attractive, conceptual objective of strategy parameter control. It is a prerequisite for a predictability of the adaptation success and enhances the transferability of performance results to real world search problems. With respect to this objective the CORR-ES (i.e. correlated mutations as typically applied [12, 13], see Sect. 3) reveals considerable deficiencies and cannot satisfy the original intention. In contrast, the so-called covariance matrix adaptation [4, 5] is invariant against any linear transformation of the search space. The disadvantage, associated with this invariance, is the possibility that the search can in principle degenerate into a subspace. The impossibility of a static degeneration in the CORR-ES must be regarded as its major advantage. It can become relevant on highly disturbed and separable objective functions.

The replicability of the results with the CORR-ES can strongly depend on the initial strategy parameters (angles), implementational details usually not explicitely defined (order of rotations) and subtle changes in the formulation of the objective function. Therefore the interpretation of results from this strategy variant must be done very carefully.

Acknowledgments

This work was supported by the *Deutsche Forschungsgemeinschaft* under grant Re 215/12-1. The author would like to thank Andreas Ostermeier and Iván Santibáñez-Koref for many essential suggestions.

References

1. Thomas Bäck, Uwe Hammel, and Hans-Paul Schwefel. Evolutionary computation: Comments on the history and current state. *IEEE Transactions on Evolutionary Computation*, 1(1):3–17, 1997.
2. Thomas Bäck and Hans-Paul Schwefel. An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation*, 1(1):1–23, 1993.
3. Nikolaus Hansen. *Verallgemeinerte individuelle Schrittweitenregelung in der Evolutionsstrategie. Eine Untersuchung zur entstochastisierten, koordinatensystemunabhängigen Adaptation der Mutationsverteilung.* Mensch und Buch Verlag, Berlin, 1998. ISBN 3-933346-29-0.
4. Nikolaus Hansen and Andreas Ostermeier. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation*, pages 312–317, 1996.
5. Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 2000. To appear.
6. Nikolaus Hansen, Andreas Ostermeier, and Andreas Gawelczyk. On the adaptation of arbitrary normal mutation distributions in evolution strategies: The generating set adaptation. In L. Eshelman, editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 57–64, Pittsburgh, 1995. Morgan Kaufmann, San Fransisco.
7. Christian Holzheuer. Analyse der Adaptation von Verteilungsparametern in der Evolutionsstrategie. Diploma thesis, Fachgebiet Bionik und Evolutionstechnik der Technischen Universität Berlin, 1996.
8. Zbigniew Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Springer-Verlag, Berlin Heidelberg New York, 1996. Third edition.
9. Ingo Rechenberg. *Evolutionsstrategie '94.* frommann-holzboog, Stuttgart, 1994.
10. Günter Rudolph. On correlated mutations in evolution strategies. In R. Männer and B. Manderick, editors, *Parallel Problem Solving from Nature, 2, Proceedings*, pages 105–114, Brüssel, 1992. North-Holland.
11. Ralf Salomon. Reevaluating genetic algorithm performance under coordinate rotation of benchmark functions. *BioSystems*, 39(3):263–278, 1996.
12. Hans-Paul Schwefel. *Numerical Optimization of Computer Models.* Wiley, Chichester, 1981.
13. Hans-Paul Schwefel. *Evolution and Optimum Seeking.* Sixth-Generation Computer Technology Series. John Wiley & Sons Inc., New York, 1995.
14. Patrick D. Surry and Nicholas J. Radcliffe. Formal algorithms + formal representations = search strategies. In H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature—PPSN IV, Proceedings*, pages 366–375, Berlin, 1996. Springer, Berlin.
15. D. Whitley, K. Mathias, S. Rana, and J. Dzubera. Evaluating evolutionary algorithms. *Artificial Intelligence*, 85:245–276, 1996.