Contemporary Evolution Strategies

Hans-Paul Schwefel[1], Günter Rudolph[2] and Thomas Bäck[2]

[1] University of Dortmund, Department of Computer Science, D-44221 Dortmund
[2] Informatik Centrum Dortmund, Center for Applied Systems Analysis (CASA),
Joseph-von-Fraunhofer-Str. 20, D-44227 Dortmund

Abstract. After an outline of the history of evolutionary algorithms, a new $(\mu, \kappa, \lambda, \rho)$ variant of the evolution strategies is introduced formally. Though not comprising all degrees of freedom, it is richer in the number of features than the meanwhile old (μ, λ) and $(\mu+\lambda)$ versions. Finally, all important theoretically proven facts about evolution strategies are briefly summarized and some of many open questions concerning evolutionary algorithms in general are pointed out.

1 A Brief History of Evolutionary Computation

Since the spring of 1993, when the first issue of the international journal on Evolutionary Computation [1] appeared, more and more people have become aware of computer algorithms known to insiders since nearly thirty years as Genetic Algorithms (GAs), Evolution Strategies (ESs), and Evolutionary Programming (EP). As a common denominator the term Evolutionary Algorithms (EAs) has become rather widespread meanwhile.

Likewise common to all three approaches to computational problem solving, mostly in terms of iterative adaptation or stepwise optimization, is the use of biological paradigms gleaned from organic evolution. Exploring the search space by means of a population of search points that underlie variation, and exploiting the information gathered in order to drive the population to ever more promising regions, are viewed as mimicking mutation, recombination, and selection.

Apart from these common features, both the independent origins and the currently used incarnations of the GA, EP, and ES algorithms differ from each other considerably. Simulating natural phenomena and processes goes back to the time when electronic automata became available to academia. Artificial neural nets, homeostatic controllers, predictors and optimizers emerged, and if their environment could not easily be modeled in simple analytical terms, random number generators became attractive in order to bridge the knowledge gap towards deciding upon the next action within the circuits. Ashby's homeostat [2], Brooks' [3] and Rastrigin's [4] optimizers are witnesses of those days.

The first digital computers, however, were not that fast as often called in papers of that time. Thus artificial or simulated evolutionary processes to solve real-world problems [5, 6] had to give way to quicker problem solving methods relying upon simple computational models like linear or quadratic input-output relations. Another reason for the dominance of greedy algorithms has been their strong backing from theory. People like to know in advance whether the (exact) solution of a problem will be found with guarantee and how many cycles of an iterative scheme are necessary to reach that goal. That is why rather often problems have been fitted to the abilities of solution algorithms by rigorous simplification.

Besides Bremermann's simulated evolution [6], used for solving nonlinear systems of equations, for example, L. Fogel [7] devised a finite state automaton for prediction tasks by simulating evolutionary mechanisms. The current version of this approach, called Evolutionary Programming (EP), is due to D. Fogel [8] and is used for continuous parameter optimization. Discrete, even combinatorial search, adaptation, and optimization is the domain of Holland's [9] Genetic Algorithm (GA) which comprises many different versions nowadays. Independently of both these origins, Evolution Strategies (ESs) came to the fore as experimental optimization techniques [10, 11], e.g., to drive a flexible pipe bending or changeable nozzle contour successively into a form with minimal loss of energy. Similar to Evolutionary Operation (EVOP [12]) the variables were changed in discrete steps, but stochastically instead of deterministically. The earliest ES version operated on the basis of two individuals only, one parent and one descendant per generation.

Theory as well as practice, more and more turning to computer simulation instead of expensive real experiments, led to multimembered ESs operating on the basis of continuous decision variables [13–17] with $\mu > 1$ parents and $\lambda > \mu$ children per cycle.

The following section introduces an even more general ES version. In section 3 a short summary will be given to what theory can tell us already about reliability and efficiency. Finally, a couple of open questions will be pointed out, and EAs in general will be turned to again.

2 The $(\mu, \kappa, \lambda, \rho)$ Evolution Strategy

Though meanwhile many, sometimes specialized, ES versions exist, we will restrict ourselves to a rather canonical set of features here. Neither parallel nor multiobjective, neither discrete nor mixed-integer special forms are considered in the following – though they exist and have been used already in applications. A recent overview may be found in [17].

In the beginning, there existed two different forms of the multimembered evolution strategy, namely the $(\mu+\lambda)$ and the (μ, λ) ESs. The symbol μ denotes the number of parents appearing at a time in a population of imaginary individuals. The symbol λ stands for the number of all offspring created by these parents within one (synchronized) generation. The difference between both approaches consists in the way the parents of a new generation are selected.

In the $(\mu + \lambda)$ ES the λ offspring and their μ parents are united, before according to a given criterion, the μ fittest individuals are selected from this set of size $\mu+\lambda$. Both μ and λ can be as small as 1 in this case, in principle. Indeed, the first experiments were all performed on the basis of a $(1 + 1)$ ES. In the

(μ, λ) ES, with $\lambda > \mu \geq 1$, the μ new parents are selected from the λ offspring only, no matter whether they surpass their parents or not. The latter version is in danger to diverge (especially in connection with self-adapting variances - see below) if the so far best position is not stored externally or even preserved within the generation cycle (so-called elitist strategy). We shall come back to that later on. So far, only empirical results have shown that the comma version has to be preferred when internal strategy parameters have to be learned online collectively. For that to work, $\mu > 1$ and intermediary recombination of the mutation variances seem to be essential preconditions. It is not true that ESs consider recombination as a subsidiary operator.

The (μ, λ) ES implies that each parent can have children only once (duration of life: one generation = one reproduction cycle), whereas in the plus version individuals may live eternally – if no child achieves a better or at least the same quality. The new $(\mu, \kappa, \lambda, \rho)$ ES introduces a maximal life span of $\kappa \geq 1$ reproduction cycles (iterations). Now, both original strategies are special cases of the more general strategy, with $\kappa = 1$ resembling the comma- and with $\kappa = \infty$ resembling the plus-strategy, respectively. Thus, the advantages and disadvantages of both extremal cases can be scaled arbitrarily. Other new options include:

- free number of parents involved in reproduction (not only 1, 2, or all)
- tournament selection as alternative to the standard (μ, λ) selection
- free probabilities of applying recombination and mutation
- further recombination types including crossover.

Though in the first ES experiments the object variables were restricted to discrete values, computerized ESs have mostly been formulated for the continuous case. An exception may be found in Rudolph [18]. The $(\mu, \kappa, \lambda, \rho)$ evolution strategy is defined here for continuous variables only by the following 19-tuple:

$$(\mu, \kappa, \lambda, \rho) ES := (P^{(0)}, \mu, \kappa, \lambda, \mathbf{rec}, p_r, \rho, \gamma, \omega, \mathbf{mut}, p_m, \tau, \tau_0, \delta, \beta, \mathbf{sel}, \zeta, t, \varepsilon) \quad (1)$$

with

$P^{(0)} := (\mathbf{a}_1, \ldots, \mathbf{a}_\mu)^{(0)} \in I^\mu$	$I := \mathbb{N}_0 \times \mathbb{R}^n \times$ $\times \mathbb{R}_+^{n_\sigma} \times [-\pi, \pi]^{n_\alpha}$	start population
$\mu \in \mathbb{N}$	$\mu \geq 1$	number of parents
$\kappa \in \mathbb{N}$	$\kappa \geq 1$	upper limit for life span
$\lambda \in \mathbb{N}$	$\lambda > \mu$ if $\kappa = 1$	number of offspring
$\mathbf{rec} : I^\mu \rightarrow I$		recombination operator
$p_r \in \mathbb{R}_+^3$	$0 \leq p_r \leq 1$	recombination probability
$\rho \in \mathbb{N}^3$	$1 \leq \rho \leq \mu$	number of ancestors for each descendant
$\gamma \in \mathbb{N}^3$	$1 \leq \gamma \leq n_x - 1$ $\gamma \geq \rho - 1$	number of crossover sites in a string of n_x elements
$\omega \in \{0, 1, 2, 3, \ldots\}^3$		type of recombination
$\mathbf{mut} : I \rightarrow I$		mutation operator
$p_m \in \mathbb{R}_+^3$	$0 < p_m \leq 1$	mutation probability

$\tau, \tau_0, \delta \in \mathbb{R}_+$	$0 \leq \delta \leq 1$	step length variabilities
$\beta \in \mathbb{R}_+$	$0 \leq \beta \leq \frac{\pi}{4}$	correlation variability
$\mathbf{sel} : I^{\mu+\lambda} \rightarrow I^{\mu}$		selection operator
$\zeta \in \mathbb{N}$	$2 \leq \zeta \leq \mu + \lambda$	tournament participators
$t : I^{2\mu} \rightarrow \{0, 1\}$		termination criterion
$\varepsilon \in \mathbb{R}_+^4$		accuracies required.

$p_r, \rho, \gamma, \omega$, and p_m may be different for object and strategy parameter variation. That is why they are introduced here as arrays of length three. Corresponding indices (x, σ, and α) have been omitted for easier reading.

The (n, f, m, G) optimization problem for continuous variables at hand may be defined as follows:

$$\text{Minimize } \{f(\mathbf{x}) \mid \mathbf{x} \in M \subseteq \mathbb{R}^n\} \tag{2}$$

with

$n \in \mathbb{N}$	dimension of the problem
$f : M \rightarrow \mathbb{R}$	objective function
$M = \{\mathbf{x} \in \mathbb{R}^n \mid g_j(\mathbf{x}) \geq 0 \;\; \forall\, j = 1, \ldots, m\}$	feasible region
$m \in \mathbb{N}_0$	number of constraints
$G = \{g_j : \mathbb{R}^n \rightarrow \mathbb{R} \;\; \forall\, j = 1, \ldots, m\}$	set of inequality restrictions.

$P^{(0)}$ denotes the initial population (iteration counter $T = 0$) of parents and consists of arbitrary vectors $\mathbf{a}_k^{(0)} \in I^{\mu}$. Each element of the population at reproduction cycle T is represented by a vector

$$\mathbf{a}_k^{(T)} = (\theta, \mathbf{x}, \sigma, \alpha)_k^{(T)} \in P^{(T)}, k \in \mathbb{N} \tag{3}$$

with

$\theta \in \mathbb{N}_0$	remaining life span in iterations (reproduction cycles), $\theta = \kappa$ at birth time
$\mathbf{x} \in \mathbb{R}^n$	vector of the object variables, the only part of $\mathbf{a}$ entering the objective function
$\sigma \in \mathbb{R}_+^{n_\sigma}$	so-called mean step sizes (standard deviations of the Gaussian distribution used for simulating mutations)
$\alpha \in [-\pi, \pi]^{n_\alpha}$	inclination angles, eventually defining (linearly) correlated mutations of the object variables $\mathbf{x}$.

The latter two vectors are called *strategy parameters* or the *internal model* of the individuals. They simply determine the variances and covariances of the n-dimensional Gaussian mutation probability density that is used for exploring the space of object variables $\mathbf{x}$.

One iteration of the strategy, that is a step from a population $P^{(T)}$ towards the next reproduction cycle with $P^{(T+1)}$, is modeled as follows:

$$P^{(T+1)} := opt_{ES}(P^{(T)}) \tag{4}$$

where $opt_{ES} : I^{\mu} \rightarrow I^{\mu}$ is defined by

$$opt_{ES} := \mathbf{sel} \circ (\mathbf{mut} \circ \mathbf{rec})^{\lambda}. \tag{5}$$

2.1 The recombination operator $\mathbf{rec}(p_r, \rho, \gamma, \omega)$

The recombination operator $\mathbf{rec} : I^{\mu} \rightarrow I$ is defined as follows:

$$\mathbf{rec} := \mathbf{re} \circ \mathbf{co} \tag{6}$$

with

$$\begin{array}{ll} \mathbf{co} : I^{\mu} \rightarrow I^{\rho} & \text{chooses } 1 \leq \rho \leq \mu \text{ parent vectors from } I^{\mu} \\ & \text{with uniform probability} \\ \mathbf{re} : I^{\rho} \rightarrow I & \text{creates one offspring vector} \\ & \text{by mixing characters from } \rho \text{ parents.} \end{array} \tag{7}$$

Depending on ω, there are several ways to recombine parents in order to get an offspring:

$\omega = 0$	no recombination; this case always holds for $\mu = 1$, $p_r = 0$, and/or $\rho = 1$
$\omega = 1$	global intermediary recombination
$\omega = 2$	local intermediary recombination
$\omega = 3$	uniform crossover
$\omega = 4$	γ point crossover.

Let $A \subseteq P^{(T)}$ of size $|A| = \rho$ be a subset of arbitrary parents chosen by the operator $\mathbf{co}$, and let $\hat{\mathbf{a}} \in I$ be the offspring to be generated. If $A = \{\mathbf{a}_1, \mathbf{a}_2\}$, $\mathbf{a}_1$ and $\mathbf{a}_2$ being two out of μ parents, holds, recombination is called *bisexual*. If $A = \{\mathbf{a}_1, \ldots, \mathbf{a}_\rho\}$ and $\rho > 2$, recombination is called *multisexual*.

Recombination in general is applied with probability p_r.

Recombination types may (often should) differ for the vectors $\mathbf{x}$ (object variables) of length n, σ (mean step sizes) of length n_σ, and α (correlation angles) of length n_α, such that $\omega \in \{0, \ldots, 4\}^3$ and $\rho \in \{1, \ldots, \mu\}^3$ have to be specified separately for these components. In the following, $\mathbf{b}$ and $\hat{\mathbf{b}}$ of lengths n_x stand for the actual parts of $\mathbf{a}$ and $\hat{\mathbf{a}}$ at hand.

The components $\hat{b}_i \;\; \forall\, i = 1, \ldots, n_x$ of $\hat{\mathbf{b}}$ are defined as follows:

$$\hat{b}_i := \begin{cases} b_i & \text{no recombination} \\ \frac{1}{\rho} \sum_{k=1}^{\rho} b_{k,i} & \text{global intermediary recombination} \\ u_i b_{k_1,i} + (1 - u_i) b_{k_2,i} & \text{local intermediary recombination,} \\ & k_1, k_2 \in \{1, \ldots, \rho\} \text{ for each offspring,} \\ & u_i \sim \mathbf{U}(0,1) \text{ or } u_i = 0.5 \\ b_{k_i,i} & \text{uniform crossover,} \\ & k_i \in \{1, \ldots, \rho\} \text{ randomly chosen for each } i \end{cases} \tag{8}$$

where $\mathbf{U}(v, w)$ denotes the uniform probability distribution with support $(v, w) \subset \mathbb{R}$. For other than uniform crossover forms, $1 \leq \gamma \leq n_x - 1$ crossover points are first chosen within the strings $\mathbf{b}_k$ at random, and then the offspring gets his $\gamma + 1$ parts of vector $\hat{\mathbf{b}}$ by turns from all of the $\rho \leq \gamma + 1$ parents involved.

2.2 The mutation operator $\mathbf{mut}(p_m, \tau, \tau_0, \delta, \beta)$

The mutation operator $\mathbf{mut} : I \rightarrow I$ is defined as follows:

$$\mathbf{mut} := \mathbf{mu}_{\mathbf{x}} \circ (\mathbf{mu}_{\sigma} \times \mathbf{mu}_{\alpha}) \tag{9}$$

with $\mathbf{mu}_{\mathbf{x}}$, $\mathbf{mu}_{\sigma}$, and $\mathbf{mu}_{\alpha}$ given below, separately.

Let $\hat{\mathbf{a}} = (\hat{\theta}, \hat{x}_1, \ldots, \hat{x}_n, \hat{\sigma}_1, \ldots, \hat{\sigma}_{n_\sigma}, \hat{\alpha}_1, \ldots, \hat{\alpha}_{n_\alpha})$ with $n, n_\sigma, n_\alpha \in \mathbb{N}$ and $n_\alpha = (n - \frac{n_\sigma}{2})\,(n_\sigma - 1)$ be the result of the recombination step. The number n denotes the problem's dimension, and $1 \leq n_\sigma \leq n$ the number of different step sizes (standard deviations for mutating the object variables $\hat{\mathbf{x}}$). It may be worthwhile to investigate the new additional degree of freedom by choosing $0 < p_m \leq 1$ for mutating the step sizes.

- $\mathbf{mu}_{\sigma} : \mathbb{R}^{n_\sigma} \rightarrow \mathbb{R}^{n_\sigma}$ mutates the recombined $\hat{\sigma}$:

$$\mathbf{mu}_{\sigma}(\hat{\mathbf{a}}) := \left(\hat{\sigma}_1 e^{z_1 + z_0}, \ldots, \hat{\sigma}_{n_\sigma} e^{z_{n_\sigma} + z_0}\right) =: \tilde{\sigma} \tag{10}$$

with

$$z_0 \sim \mathbf{N}(0, \tau_0^2), \quad z_i \sim \mathbf{N}(0, \tau^2) \;\; \forall\, i = 1, \ldots, n_\sigma. \tag{11}$$

$\mathbf{N}(\xi, \psi^2)$ denotes the normal distribution with mean ξ and variance ψ^2 (standard deviation ψ). For maximal rates of convergence in case of the so-called sphere model (see section 3), τ and τ_0 may be chosen according to the relationships:

$$\tau_0 = \frac{K}{\sqrt{p_m}} \frac{\delta}{\sqrt{n}}, \quad \tau = \frac{K}{\sqrt{p_m}} \frac{\sqrt{1 - \delta^2}}{\sqrt{\frac{n}{\sqrt{n_\sigma}}}} \tag{12}$$

where the constant K should reflect the convergence velocity of the ES (see section on theoretical results). So far, only $\delta = \frac{1}{\sqrt{2}}$ and $p_m = 1$ have been used, but other values may be worthwhile to be considered as well.

- $\mathbf{mu}_{\alpha} : \mathbb{R}^{n_\alpha} \rightarrow \mathbb{R}^{n_\alpha}$ mutates the recombined $\hat{\alpha}$:

$$\mathbf{mu}_{\alpha}(\hat{\mathbf{a}}) := (\hat{\alpha}_1 + z_1, \ldots, \hat{\alpha}_{n_\alpha} + z_{n_\alpha}) =: \tilde{\alpha} \tag{13}$$

with

$$z_i \sim \mathbf{N}(0, \beta^2) \;\; \forall\, i = 1, \ldots, n_\alpha. \tag{14}$$

Good results have been obtained with $\beta \approx 0.0873\,[\approx 5^\circ\,]$, but the question whether β should be different for each z_i is still open.

- $\mathbf{mu}_{\mathbf{x}}(\tilde{\sigma}, \tilde{\alpha}) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ mutates the recombined object variables $\hat{\mathbf{x}}$, using the recombined and already mutated $\tilde{\sigma}, \tilde{\alpha}$ (for efficiency reasons only, otherwise the sequence of all variation steps and even of the cyclical selection and variation processes does not matter):

$$\mathbf{mu_x}(\tilde{\sigma},\tilde{\alpha})(\hat{\mathbf{a}}) := (\hat{x}_1 + cor_1(\tilde{\sigma},\tilde{\alpha}),\ldots,\hat{x}_n + cor_n(\tilde{\sigma},\tilde{\alpha})) =: \tilde{\mathbf{x}} \tag{15}$$

where $\mathbf{cor} := (cor_1,\ldots,cor_n)$ is a random vector with normally distributed, eventually correlated components, using $\tilde{\sigma}$ and $\tilde{\alpha}$. The components of $\mathbf{cor}$ can be calculated as follows [19]:

$$\mathbf{cor} = \mathbf{T}\,\mathbf{z} \tag{16}$$

where $\mathbf{z} = (z_1,\ldots,z_{n_\sigma})$ with $z_i \sim \mathbf{N}(0,\tilde{\sigma}_i^2)\ \ \forall\, i = 1,\ldots,n_\sigma$ and

$$\mathbf{T} = \prod_{p=1}^{n_\sigma - 1} \prod_{q=p+1}^{n_\sigma} \mathbf{T}_{pq}(\tilde{\alpha}_j) \tag{17}$$

with $j = \frac{1}{2}(2n_\sigma - p)(p+1) - 2n_\sigma + q$ and

$$\mathbf{T}_{pq}(\tilde{\alpha}_j) := \begin{pmatrix}
1 & 0 & & & & & \cdots & & & & & 0 & \\
0 & 1 & & & & & & & & & & & \\
 & & \ddots & & & & & & & & & & \\
 & & & 1 & & & & & & & & & \\
 & & & & \cos\,\tilde{\alpha}_j & & & & -\sin\,\tilde{\alpha}_j & & & & \\
 & & & & & 1 & & & & & & & \\
\vdots & & & & & & \ddots & & & & & & \vdots \\
 & & & & & & & 1 & & & & & \\
 & & & & \sin\,\tilde{\alpha}_j & & & & \cos\,\tilde{\alpha}_j & & & & \\
 & & & & & & & & & 1 & & & \\
 & & & & & & & & & & \ddots & & \\
 & & & & & & & & & & & 1 & 0 \\
0 & & & & & & \cdots & & & & & 0 & 1
\end{pmatrix} \tag{18}$$

with the terms $\cos\,\tilde{\alpha}_j$ and $\pm\sin\,\tilde{\alpha}_j$ in columns p and q and lines p and q, respectively. An efficient way of calculating (16) is the multiplication from right to left. $\tilde{\theta}$ is set to $\tilde{\theta} = \kappa$ for all offspring when they are created by means of recombination and mutation. Finally we have:

$$\tilde{\mathbf{a}}_k = (\tilde{\theta},\tilde{\mathbf{x}},\tilde{\sigma},\tilde{\alpha})\ \ \forall\, k = 1,\ldots,\lambda. \tag{19}$$

For constrained optimization the processes of recombination and mutation must be repeated as often as necessary to create λ non-lethal offspring such that

$$g_j(\tilde{\mathbf{x}}_k)\ \geq\ 0\ \ \forall\, j = 1,\ldots,m \text{ and }\ \forall\, k = 1,\ldots,\lambda. \tag{20}$$

This vitality check may already be part of the selection process, however.

2.3 The selection operator $\mathbf{sel}(\zeta)$

Natural selection is a term that tries to describe the final result of several different real world processes, i.e., from the test of new born individuals against natural laws (if not met, the trial is lethal) and other environmental conditions up to what is called mating selection. According to Darwin, selection mainly helps to avoid the Malthusian trap of food shortage due to overpopulation, the result of a normal surplus of births over deaths (this is neither reflected in GAs nor in EP). Others emphasize the reproduction success of stronger or more intelligent individuals, perhaps induced by Darwin's unlucky term 'struggle for life.'

Altogether, there are several ways of implementing selection mechanisms. Two typical selection operators will be presented here. They mainly differ in the selection pressure they exert on a population. Due to the strong impact of selection on the behavior of the evolutionary process, it is worthwhile to provide both schemes.

The *traditional deterministic ES selection operator* can be defined as:

$$\mathbf{sel} : I^{\mu+\lambda} \rightarrow I^{\mu}. \tag{21}$$

Let $P^{(T)}$ denote some parent population in reproduction cycle T, $\tilde{P}^{(T)}$ their offspring produced by recombination and mutation, and $Q^{(T)} = P^{(T)} \sqcup \tilde{P}^{(T)} \in I^{\mu+\lambda}$ where the operator $\sqcup$ denotes the union operation on multisets. Then

$$P^{(T+1)} := \mathbf{sel}(Q^{(T)}). \tag{22}$$

The next reproduction cycle contains the μ best individuals, i.e., the following relation is valid:

$$\forall\, \mathbf{a} \in P^{(T+1)} : \theta_a > 0 \ \wedge \ \nexists\, \mathbf{b} \in Q^{(T)} \setminus P^{(T+1)} : \mathbf{b} \overset{\kappa}{>} \mathbf{a} \tag{23}$$

where the relation $\overset{\kappa}{>}$ (read: better than) introduces a maximum duration of life, κ, that defines an individual to be worse than an other one if its age is greater than the allowed maximum, κ, or if its fitness (measured by the objective function) is worse.

The definition of the $\overset{\kappa}{>}$ - relation is given by:

$$\mathbf{a}_k \overset{\kappa}{>} \tilde{\mathbf{a}}_\ell :\Leftrightarrow \theta_k > 0 \ \wedge \ f(\mathbf{x}_k) \leq f(\tilde{\mathbf{x}}_\ell). \tag{24}$$

In practical applications, where constraints can be evaluated quickly (e.g., in the case of simple bounds to the object variables), it may be advantageous to evaluate the constraints first. Thus, only if a search point lies within the feasible region (non-lethal individual), the time consuming objective function has to be evaluated. However, things may turn out to be just the other way round, i.e., the time consuming part of the evaluation lies in the check for feasibility (e.g., if a FEM is used to calculate the stresses and deformations of a mechanical structure, the result of which must be compared with given upper bounds). Then the selection process must be interwoven with the process of generating offspring by recombination and mutation.

At the end of the selection process, the remaining maximum life durations have to be decremented by one for each survivor:

$$\theta_k^{(T+1)} := \tilde{\theta}_k^{(T)} - 1 \;\; \forall \; k = 1, \ldots, \mu. \tag{25}$$

The *tournament selection* is well suited for parallelization of the selection process. This method selects μ times the best individual from a random subset B_k of size $|B_k| = \zeta$, $2 \leq \zeta \leq \mu + \lambda \;\; \forall \; k = 1, \ldots, \mu$ and transfers it to the next reproduction cycle (note that there may appear duplicates!). The best individual within each subset B_k is selected according to the $\overset{\kappa}{>}$ relation which was introduced in (24). A formal definition of the $(\mu, \kappa, \lambda, \zeta)$ tournament selection follows:
Let

$$B_k \subseteq Q^{(T)} \;\; \forall \; k = 1, \ldots, \mu \tag{26}$$

be random subsets of $Q^{(T)}$, each of size $|B_k| = \zeta$. For each $k \in \{1, \ldots, \mu\}$ choose $\mathbf{a}_k \in B_k$ such that

$$\forall \; \mathbf{b} \in B_k : \mathbf{a}_k \overset{\kappa}{>} \mathbf{b} \; . \tag{27}$$

Finally,

$$P^{(T+1)} := \bigsqcup_{k=1}^{\mu} \{ \, \mathbf{a}_k^{(T+1)} \, \}. \tag{28}$$

2.4 The termination criterion $t(\varepsilon)$

The termination of the new evolution strategy should be handled in the same way as has been done within the older versions [17]:

All digital computers handle data only in the form of a finite number of units of information (bits). The number of significant figures and the range of numbers is thereby limited. If a quantity is repeatedly divided by a factor greater than one, the stored value of the quantity eventually becomes zero after a finite number of divisions. Every subsequent multiplication leaves the value as zero. If this happens to one of the standard deviations σ_i, the affected variable x_i remains constant thereafter. The optimization continues only in a subspace of $\mathbb{R}^n$. To guard against this it must be required that $\sigma_i > 0 \;\; \forall \; i = 1, \ldots, n_\sigma$. The random changes should furthermore be sufficiently large that at least the last stored digit of a variable is altered. There are therefore two requirements:

Lower limits for the “step lengths”:

$$\sigma_i^{(T)} \geq \varepsilon_1 \;\; \forall \; i = 1, \ldots, n_\sigma \tag{29}$$

and

$$\sigma_i^{(T)} \geq \varepsilon_2 \left| x_i^{(T)} \right| \;\; \forall \; i = 1, \ldots, n_\sigma \tag{30}$$

where

$$\left. \begin{array}{r} \varepsilon_1 > 0 \\ 1 + \varepsilon_2 > 1 \end{array} \right\} \text{ according to the computational accuracy} \tag{31}$$

It is thereby ensured that the random variations are always active and the region of the search stays spanned in all dimensions. Proper values for ε_1 and ε_2 may be obtained automatically by means of a small subroutine in order to suit to the computer used.

From the population of μ parents with $\mathbf{x}_k \;\; \forall\, k = 1, \ldots, \mu$, let f_b be the best objective function value:

$$f_b = \min_k \{ f(\mathbf{x}_k^{(T)}) \;\; \forall\, k = 1, \ldots, \mu \} \tag{32}$$

and f_w the worst:

$$f_w = \max_k \{ f(\mathbf{x}_k^{(T)}) \;\; \forall\, k = 1, \ldots, \mu \} \tag{33}$$

Then for ending the search we require that either

$$f_w - f_b \leq \varepsilon_3 \tag{34}$$

or

$$\frac{\mu}{\varepsilon_4} (f_w - f_b) \leq \sum_{k=1}^{\mu} \left| f(\mathbf{x}_k^{(T)}) \right| =: f_m \tag{35}$$

where ε_3 and ε_4 are to be defined such that (compare with ε_1 and ε_2 above)

$$\left. \begin{array}{r} \varepsilon_3 > 0 \\ 1 + \varepsilon_4 > 1 \end{array} \right\} \text{ according to the computational accuracy} \tag{36}$$

Either absolutely or relatively, the objective function values of the parents existing at a time must fall closely together before termination is stated.

An other possibility to decide upon termination is to look for the whole progress of the population during a certain number of iterations. This can be based on the current best (f_b) or (f_m) value of the objective function, e.g.:

Stop the optimum seeking process if

$$f_y^{(T-\Delta T)} - f_y^{(T)} \leq \varepsilon_3 \tag{37}$$

or

$$\frac{2}{\varepsilon_4} (f_y^{(T-\Delta T)} - f_y^{(T)}) \leq \left| f_y^{(T-\Delta T)} \right| + \left| f_y^{(T)} \right| \tag{38}$$

where f_y denotes either f_b or f_m as defined in 35. The threshold ΔT could depend on n, the number of variables.

2.5 The start conditions $P^{(0)}$

For reasons of comparability with GAs on the one hand and more classical optimization techniques on the other, there should be two distinct ways of setting up the initial population.

Case a: With given lower and upper bounds for all object variables (a prerequisite for all GAs, but not for ESs)

$$\underline{x}_i \leq x_i \leq \bar{x}_i \quad \forall\, i = 1, \ldots, n \tag{39}$$

all μ parents at cycle $T = 0$ are arbitrarily distributed within the bounded region.

Case b: With a given start position $\mathbf{x}^{(0)}$ for the optimum seeking process that is assigned to one individual $\mathbf{a}_1$, the other $\mu - 1$ parents for the first iteration cycle are found by applying some kind of mutation process with enlarged step sizes $c\,\sigma^{(0)}, c > 1$, for example $c = 10$, by:

$$x_{k,i}^{(0)} = x_{1,i}^{(0)} + c\sigma_i^{(0)} z_i \text{ with } z_i \sim \mathbf{N}(0,1) \;\; \forall\, i = 1, \ldots, n \text{ and } \forall\, k = 2, \ldots, \mu. \tag{40}$$

One may increase c during this setup process if no constraints are violated and the objective function has been improved during the last step; otherwise c should be decreased.

2.6 The handling of constraints

During the optimum seeking process of ESs, inequality constraints so far have been handled as barriers, i.e., offspring that violate at least one of the restrictions are lethal mutations. Before the selection operator can be activated, exactly λ non-lethal offspring must have been generated.

In case of a non-feasible start position $\mathbf{x}^{(0)}$, a feasible solution must be found at first. This can be achieved by means of an auxiliary objective function

$$\tilde{f}(\mathbf{x}) = \sum_{j=1}^{m} g_j(\mathbf{x})\,\delta_j(\mathbf{x}) \tag{41}$$

with $\delta_j(\mathbf{x}) = -1$ if $g_j(\mathbf{x}) < 0$ and zero otherwise. Each decrease of the value of $\tilde{f}(\mathbf{x})$ represents an approach to the feasible region. As soon as $\tilde{f}(\mathbf{x}) = 0$ can be stated, then $\mathbf{x}$ satisfies all the constraints and can serve as a starting vector for the optimization proper.

3 Theoretical Results

All theoretical results heavily rely upon simplifications of the situation investigated, here with respect to the objective functions taken into consideration, as well as with respect to the optimum seeking algorithm. The gap between practical results in very difficult situations – so far these happen to be the sole justification for EAs – and theoretically proven results in rather simple situations – in which normally other solution techniques are preferable – remains huge. Nevertheless, an algorithm that is worth consideration for complex tasks should fulfill some minimal requirements in easy situations as well.

Global convergence in case of nearly arbitrary response surface landscapes (called effectivity or robustness) and high efficiency, i.e., a low number of function evaluations until achieving a specified approximation to the exact solution are maximal requirements that will remain in conflict with one another, forever. EAs have often been apostrophed as universal search methods since they do not try to gain advantage from higher order information like derivatives of the objective function and do not interpret intermediate results on the basis of a specific (e.g., linear or quadratic) internal model of the fitness landscape. As minimal requirements to them one must demand that they should never fail in simple cases and that they should provide some means to scale their behavior towards maintaining usefulness in more difficult situations, e.g., by choosing an appropriate population size and selection pressure. As long as no superior method is available, an EA arriving at one of the better local, though not global, optima is a useful search method.

Up until now, no all-embracing theory for the $(\mu, \kappa, \lambda, \rho)$ ES exists. However, some special cases like $\kappa = 1$ or $\kappa = \infty, \rho = 1$ or $\rho = \mu$, and $\mu = 1$ have been investigated thoroughly. The results will be summarized briefly in the following.

A very early global convergence proof for a $(1+1)$ ES with one parent and one descendant per generation, elitist selection, no recombination and normally distributed mutations without correlation has been given by Born [20]. No continuity, differentiability, or unimodality assumptions must be made. Except for singular solutions, global convergence with probability one in the limit of infinitely many mutations is guaranteed as long as the mutation variance is greater than zero in all directions. The same holds for the more general $(\mu + \lambda)$ ES [21]. More delicate is the non-elitist case of a $(1, \lambda)$ ES for which Rudolph [22] has developed sufficient conditions under which convergence is maintained. Like the canonical GA, the $(1, \lambda)$ ES with fixed mutation variances finally stagnates at a distance from the optimum that depends on σ (actually, it fluctuates around that position).

More interesting than all that is an answer to the question of the approximation velocity. Whereas this question is still open for GAs, except for a very special case (see [23]), the situation is somewhat better for ESs now. Linear convergence order, that is a constant increase of accurate figures of the object parameter or objective function values over the number of mutations or generations, has been proved by Rappl [24] for the $(1+1)$ ES when applied to a strongly convex fitness function like

$$f_1(\mathbf{x}) = c_0 + \sum_{i=1}^{n} c_i (x_i - x_i^*)^2 \tag{42}$$

which has been called spherical model if $c_i = 1 \;\; \forall i = 1, \ldots, n$. Most of the following results are valid not only for this spherical situation, but also for functions like $f_2(\mathbf{x}) = -e^{-f_1(\mathbf{x})}$ (a nightmare for quasi-Newton methods, which diverge everywhere in this case) and, approximately at least, also in case of higher even exponents than two in function f_1 [17].

One basic assumption for the proof has been the maintenance of the corresponding optimal mutation variances. For the (1+1) ES this can be approximately achieved by applying the so-called 1/5 success rule. Following Rechenberg [13] the mutation variance should be increased as soon as the observed success rate is greater than 1/5 and decreased if it turns out to be less than 1/5 (for a critique, see [17]). This result was achieved for the spherical function above and for an endless inclined ridge model with rectangular cross-section of constant width perpendicular to the steepest descent direction, which is the main diagonal in case of

$$f_3(\mathbf{x}) = c_0 - \sum_{i=1}^{n} c_i x_i \tag{43}$$

if again $c_i = 1 \;\; \forall i = 1, \ldots, n$.

This is not the place to go into more details, but proportional control according to that rule has proven to lead to oscillatory behavior with some factor loss in convergence velocity against the optimal case. Linear convergence order, however, is maintained.

Rechenberg [16] claims linear convergence order for the (μ, λ) ES on the sphere model in case of optimal mutation variances. His law

$$\varphi = E\{\Delta r\} = C\sigma - \frac{n}{2r}\sigma^2 \tag{44}$$

for the expected difference $\Delta r = r^{(g-1)} - r^{(g)}$ between the Euclidean distances $r = r^{(g-1)}$ before and $r^{(g)}$ after one generation is an approximation to the very high-dimensional case ($n \gg 1$), as Beyer [25] has elaborated. In terms of the dimensionless quantities $\breve{\varphi} = \frac{n}{r}\varphi$ and $\breve{\sigma} = \frac{n}{r}\sigma$ the law reads

$$\breve{\varphi} = C\breve{\sigma} - \frac{1}{2}\breve{\sigma}^2. \tag{45}$$

The maximal convergence rate $\varphi^* = \breve{\varphi}_{max} = \frac{1}{2}C^2$ corresponds to the optimal standard deviation $\sigma^* = \breve{\sigma}_{opt} = C$, a constant that of course still depends on $\mu, \kappa, \lambda,$ and ρ, at least.

An approximation with respect to λ, when $\mu, \kappa,$ and ρ equal one, has been found to be as simple as [21]:

$$C \approx \sqrt{2 \ln \lambda}. \tag{46}$$

Beyer [26] has investigated both uniform crossover and global intermediary multi-recombination with $\rho = \mu$. Though the laws (again approximations) look different

$$\breve{\varphi} = C\breve{\sigma} - \frac{1}{2\mu}\breve{\sigma}^2 \quad \text{(global intermediary)} \tag{47}$$

$$\breve{\varphi} = \sqrt{\mu}\, C\breve{\sigma} - \frac{1}{2}\breve{\sigma}^2 \quad \text{(uniform crossover)} \tag{48}$$

they yield the same maximum $\varphi^* = \frac{\mu}{2}\, C^2$ for $\sigma^* = \mu\, C$ in the former, $\sigma^* = \sqrt{\mu}\, C$ in the latter case. If one interprets φ as an approximation to the differential $\dot{r} = dr/dt$ (literally: the velocity of approaching the optimum $\mathbf{x}^*$ of the sphere model function [17]) one arrives at concluding $K = \varphi^*$ for the still open constant in relation (12). Based upon the observation that in the linear theory the convergence rate mainly depends on the ratio $\frac{\lambda}{\mu}$ if the population size is not too small, one might speculate about putting together what we know so far to the rather simple formula (for large n and λ, as well as not too small μ):

$$\varphi^* \sim \mu \ln \frac{\lambda}{\mu} \tag{49}$$

the first factor (μ) being due to the diversity of the population and exploited by recombination (so-called genetic repair [26]), the latter ($\frac{\lambda}{\mu}$) due to the selection pressure - two processes that compete with one another. A proof is still missing, however. The same holds for the influence of other strategy parameters like κ, p_r, and p_m, which were assumed to equal one in the considerations above.

Of great practical interest is the answer to the question how to achieve and maintain the optimal mutation variance σ^{*2}. The algorithm presented in section 2 tries to do it in a way called self-adaptation. It even allows for on-line learning of up to n different σ_i and $\frac{n}{2}(n-1)$ different α_j, thus presenting the ultimate degree of freedom for normally distributed mutations. No theory is available for that process, only a few experiments [27] have demonstrated that self-adaptation is possible under certain conditions. If these conditions are not observed the process fails and the resulting ES may not converge, even diverge in case of small values for κ.

The necessity of individually scaled σ_i can easily be derived from function f_1 in case of non-identical coefficients c_i. Nobody should be astonished that superfluous degrees of freedom, e.g., individual σ_i for the spherical model, where $\sigma_i = \sigma_0 \;\forall\, i = 1, \ldots, n$ would be the best choice, come at an extra cost. Introducing correlated mutations ($\alpha_j \neq 0$), however, does not slow down the progress further in this case. Correlations, properly learned, would help a lot in case of a hyperelliptical scene with main axes different from the coordinate axes like

$$f_4(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2 . \tag{50}$$

This still simple quadratic function poses heavy difficulties to all optimization algorithms that rely upon decomposability. Both the so-called coordinate (or Gauss-Seidel, or one-variable-at-a-time) search technique and many GA variants come into trouble with f_4. Major improvements require *simultaneous* changes of many if not all x_i. Correlated mutations help, but discrete recombination of the object variables turns out to be disastrous in this case. Why? Without correlation and independent σ_i, the population of an ES tends to concentrate at one out of two positions where the curvature radii are smallest. With individual σ_i and correlation it spreads more or less to the whole temporal hypersurface $f(x^{(T)}) = const.$ Discrete recombination then often fails to produce descendants which keep to the vicinity of that hypersurface. Intermediary recombination of the object variables, even merely switching off recombination, heals that. That is why the recombination type and frequency should be incorporated into the set of internal strategy parameters, too. In nature there is no higher instance for controlling internal parameters in the way which has been proposed with the so-called nested or meta-evolution approach [16]. More natural seem to be simulations with several subpopulations, a concept that has been used for EA incarnations on parallel computers [28].

References

1. De Jong, K. (Ed.) (1993), Evolutionary computation (journal), MIT Press, Cambridge MA
2. Ashby, W.R. (1960), Design for a brain, 2nd ed., Wiley, New York
3. Brooks, S.H. (1958), A discussion of random methods for seeking maxima, Oper. Res. **6**, 244-251
4. Rastrigin, L.A. (1960), Extremal control by the method of random scanning, Automation and Remote Control **21**, 891-896
5. Favreau, R.F., R. Franks (1958), Random optimization by analogue techniques, Proceedings of the IInd Analogue Computation Meeting, Strasbourg, Sept. 1958, pp. 437-443
6. Bremermann, H.J. (1962), Optimization through evolution and recombination, in: Yovits, M.C., G.T. Jacobi, D.G. Goldstein (Eds.), Self-organizing systems, Spartan, Washington, DC, pp. 93-106
7. Fogel, L.J. (1962), Autonomous automata, Ind. Research **4**, 14-19
8. Fogel, D.B. (1995), Evolutionary Computation—toward a new philosophy of machine intelligence, IEEE Press, Piscataway NJ
9. Holland, J.H. (1975), Adaptation in natural and artificial systems, University of Michigan Press, Ann Arbor MI
10. Rechenberg, I. (1964), Cybernetic solution path of an experimental problem, Royal Aircraft Establishment, Library Translation 1122, Farnborough, Hants, Aug. 1965, English translation of the unpublished written summary of the lecture "Kybernetische Lösungsansteuerung einer experimentellen Forschungsaufgabe", delivered at the joint annual meeting of the WGLR and DGRR, Berlin, 1964
11. Klockgether, J., H.-P. Schwefel (1970), Two-phase nozzle and hollow core jet experiments, in: Elliott, D.G. (Ed.), Proceedings of the 11th Symposium on Engineering Aspects of Magnetohydrodynamics, Caltech, March 24-26, 1970, California Institute of Technology, Pasadena CA, pp. 141-148
12. Box, G.E.P. (1957), Evolutionary operation—a method for increasing industrial productivity, Appl. Stat. **6**, 81-101
13. Rechenberg, I. (1973), Evolutionsstrategie—Optimierung technischer Systeme nach Prinzipien der biologischen Evolution, Frommann-Holzboog, Stuttgart
14. Schwefel, H.-P. (1977), Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie, Birkhäuser, Basle, Switzerland
15. Schwefel, H.-P. (1981), Numerical optimization of computer models, Wiley, Chichester
16. Rechenberg, I. (1994), Evolutionsstrategie '94, Frommann-Holzboog, Stuttgart
17. Schwefel, H.-P. (1995), Evolution and optimum seeking, Wiley, New York
18. Rudolph, G. (1994), An evolutionary algorithm for integer programming, in: Davidor, Y., H.-P. Schwefel, R. Männer (Eds.), Parallel problem solving from nature

3, Proceedings of the 3rd PPSN Conference, Jerusalem, Oct. 9-14, 1994, vol. 866 of Lecture Notes in Computer Science, Springer, Berlin, pp. 139-148

19. Rudolph, G. (1992), On correlated mutation in evolution strategies, in: Männer, R., B. Manderick (Eds.), Parallel problem solving from nature 2, Proceedings of the 2nd PPSN Conference, Brussels, Sept. 28-30, 1992, North-Holland, Amsterdam, pp. 105-114
20. Born, J. (1978), Evolutionsstrategien zur numerischen Lösung von Adaptationsaufgaben, Dr. rer. nat. Diss., Humboldt University at Berlin
21. Bäck, T., G. Rudolph, H.-P. Schwefel (1993), Evolutionary programming and evolution strategies—similarities and differences, in: Fogel, D.B., J.W. Atmar (Eds.), Proceedings of the 2nd Annual Conference on Evolutionary Programming, San Diego, Feb. 25-26, 1993, Evolutionary Programming Society, La Jolla CA, pp. 11-22
22. Rudolph, G. (1994), Convergence of non-elitist strategies, in: Michalewicz, Z., J.D. Schaffer, H.-P. Schwefel, and D.B. Fogel (Eds.), Proceedings of the 1st IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, Orlando FL, June 27-29, 1994, vol. 1, pp. 63–66
23. Bäck, T. (1994), Evolutionary algorithms in theory and practice, Dr. rer. nat. Diss., University of Dortmund, Department of Computer Science, Feb. 1994
24. Rappl, G. (1984), Konvergenzraten von Random-Search-Verfahren zur globalen Optimierung, Dr. rer. nat. Diss., Hochschule der Bundeswehr, Munich-Neubiberg, Department of Computer Science, Nov. 1984
25. Beyer, H.-G. (1995), Toward a theory of evolution strategies—the (μ, λ)-theory, submitted to Evolutionary Computation
26. Beyer, H.-G. (1994), Towards a theory of 'evolution strategies'—results from the N-dependent (μ, λ) and the multi-recombinant $(\mu/\mu, \lambda)$ theory, technical report SYS-5/94, Systems Analysis Research Group, University of Dortmund, Department of Computer Science, Oct. 1994
27. Schwefel, H.-P. (1987), Collective phenomena in evolutionary systems, in: Checkland, P., I. Kiss (Eds.), Problems of constancy and change—the complementarity of systems approaches to complexity, papers presented at the 31st Annual Meeting of the International Society for General System Research, Budapest, Hungary, June 1-5, International Society for General System Research, vol. 2, pp. 1025-1033
28. Rudolph, G. (1991), Global optimization by means of distributed evolution strategies, in: Schwefel, H.–P. and R. Männer (Eds.), Parallel problem solving from nature, Proceedings of the 1st PPSN Conference, Dortmund, Oct. 1-3, 1990, vol. 496 of Lecture Notes in Computer Science, Springer, Berlin, pp. 209-213

This article was processed using the LaTeX macro package with LLNCS style