Neutrality: A Necessity for Self-Adaptation

Marc Toussaint Christian Igel
Institut für Neuroinformatik, Ruhr-Universität Bochum
44780 Bochum, Germany
{ Marc.Toussaint,Christian.Igel }@neuroinformatik.ruhr-uni-bochum.de

Abstract—Self-adaptation is used in all main paradigms of evolutionary computation to increase efficiency. We claim that the basis of self-adaptation is the use of neutrality. In the absence of external control neutrality allows a variation of the search distribution without the risk of fitness loss.

I. Introduction

It is well known that the genotype-phenotype mapping in natural evolution is highly redundant (i.e., the mapping is not injective) and that neutral variations frequently occur. (A variation is called neutral if it alters the genotype but not the phenotype of an individual.) The potential positive effects of neutrality have extensively been discussed [6, 12, 14, 24] and there is growing interest in investigating how these results from biology carry over to evolutionary algorithms (EAs) [5, 7, 13, 17, 27, 28, 32]. However, this dissents with the opinion that redundant encodings are inappropriate for *real-world* problems—a bijective genotype-phenotype mapping is often regarded as a design principle for efficient EAs (e.g., see [20]).

Unlike neutrality, self-adaptation is an established key concept in evolutionary computation [4]. We will point out that neutrality is a necessity for self-adaptation. And since there is no doubt about the benefits of self-adaptation for a wide range of problems, this is a strong argument for the usefulness of neutral encodings. Our point of view is that, in the absence of external control, neutrality provides a way to vary the search distribution without the risk of fitness loss. We propose to term this way of adapting the search strategy self-adaptation. This approach generalizes standard definitions and underlines the central role of neutrality.

We reconsider two instructive examples from the literature: The first example is from molecular biology, where it is shown how neutrality can increase the variability of certain regions in the genome and conserve the information in other regions. Using our definition, this is self-adaptation. The second one deals with self-adaptation in evolution strategies, a well-known example of successful self-adaptation in evolutionary computation. These algorithms rely on neutrality to adapt the search strategy and are frequently applied for solving real-world optimization tasks.

Our arguments are based on recent efforts to provide a unifying view on the search distribution in evolutionary algorithms and its (self-)adaptation [31]. We introduce this formalism in the following section. Section III describes the relation between self-adaptation and neutrality and in section IV we present the two examples to illustrate our ideas.

II. Modeling evolutionary exploration

In this section, we outline that the individuals in the population, the genotype-phenotype mapping and the variation operators including their parameters can be regarded as a parameterization of the search distribution. This is done in the framework of global random search and evolutionary optimization, although all considerations can be transfered to adaptation in natural evolution, see Sec. III-B.

Evolutionary algorithms can be considered as a certain class of global random search algorithms. Let the search problem under consideration be described by a quality function $\Phi : \mathcal{P} \to \mathbb{R}$ to be optimized. The set $\mathcal{P}$ denotes the search space. According to [33], the general scheme of global random search is given by:[1]

1. Choose a probability distribution $P^{(t)}_{\mathcal{P}}$ on $\mathcal{P}$.
2. Obtain points $\boldsymbol{g}^{(t)}_1, \dots, \boldsymbol{g}^{(t)}_\lambda$ by taking λ samples from the distribution $P^{(t)}_{\mathcal{P}}$. Evaluate Φ (perhaps with random noise) at these points.
3. According to a fixed (algorithm dependent) rule construct a probability distribution $P^{(t+1)}_{\mathcal{P}}$ on $\mathcal{P}$.
4. Check for some appropriate stopping condition; if the algorithm has not terminated, substitute $t := t+1$ and return to Step 2.

The core ingredient of this search scheme is the search distribution $P^{(t)}_{\mathcal{P}}$, also called exploration distribution, on the search space. Random search algorithms can differ fundamentally in the way they represent and alter the search distribution. Typically, the distribution is represented by a semi-parametric model. The choice of the model determines the exploration distributions the search algorithm can represent, which is in general only a small subset of all possible probability distributions on $\mathcal{P}$. For example, let $\mathcal{P} = \mathbb{R}$. Then the class of repre-

[1] This scheme does not account for all kinds of evolutionary computation. For example, if the evolutionary programming scheme is used, where each parent generates exactly one offspring [9], or recombination is employed, then the EA is better described as operating on a search distribution $P^{(t)}_{\mathcal{P}^\lambda}$ on $\mathcal{P}^\lambda$. However, in these cases $P^{(t)}_{\mathcal{P}}$ can still be derived from $P^{(t)}_{\mathcal{P}^\lambda}$.

sentable distributions may be given by the class of normal densities, $p_{\mathcal{P}}(x; m, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x-m)^2}{2\sigma^2}\right\}$, where m is the expectation and σ^2 the variance. This equation defines a *parameterization* of the search distribution, it maps the parameters $(m, \sigma) \in \mathbb{R}^2$ into the set of probability distributions on $\mathcal{P}$ (see Fig. 2). A global random search algorithm alters its exploration distribution (see Step 3) by changing such parameters.

We believe that two of the most fundamental questions in evolution theory can be stated as:

1. How is the exploration distribution parameterized?
2. How is the exploration distribution altered?

In the framework of evolutionary systems, we identify each element of $\mathcal{P}$ with certain phenotypic traits of individuals and call $\mathcal{P}$ the phenotype space. Each phenotype p, i.e., element of $\mathcal{P}$, is encoded by a genotype $\boldsymbol{g}$, which is an element of a genotype space $\mathcal{G}$. The mapping $\phi : \mathcal{G} \to \mathcal{P}$ is called genotype-phenotype mapping. If ϕ is not injective we speak of a *neutral* encoding. A number of genotypes $\tilde{\boldsymbol{g}}_1^{(t)}, \dots, \tilde{\boldsymbol{g}}_\mu^{(t)}$, the parents, are stored in a population. The superscript indicates the iteration of the algorithm, i.e., the generation. In each generation, λ offspring $\boldsymbol{g}_1^{(t)}, \dots, \boldsymbol{g}_\lambda^{(t)}$ are generated by applying stochastic and / or deterministic variation operators. Let the probability that parents $\tilde{\boldsymbol{g}}_1, \dots, \tilde{\boldsymbol{g}}_\mu \in \mathcal{G}$ generate an offspring $\boldsymbol{g} \in \mathcal{G}$ be described by the probability distribution $P_{\mathcal{G}}\left(\boldsymbol{g} \mid \tilde{\boldsymbol{g}}_1, \dots, \tilde{\boldsymbol{g}}_\mu; \boldsymbol{\theta}\right)$. This distribution is additionally parameterized by some *external strategy parameters* $\boldsymbol{\theta} \in \Theta$. Examples of such externally controlled parameters include the probabilities that certain variation operators are applied and parameters that determine the mutation strength. We call the probability

$$P_{\mathcal{G}}^{(t)}(\boldsymbol{g}) := P_{\mathcal{G}}\left(\boldsymbol{g} \mid \tilde{\boldsymbol{g}}_1^{(t)}, \dots, \tilde{\boldsymbol{g}}_\mu^{(t)}; \boldsymbol{\theta}^{(t)}\right) \tag{1}$$

that in generation t an offspring $\boldsymbol{g}$ is created the *exploration distribution* on $\mathcal{G}$ at generation t.

The genotype-phenotype mapping ϕ *lifts* $P_{\mathcal{G}}^{(t)}$ from the genotype space onto the phenotype space:

$$\forall p \in \mathcal{P} : \; P_{\mathcal{P}}^{(t)}(p) = \sum_{\boldsymbol{g}' \in \phi^{-1}(p)} P_{\mathcal{G}}^{(t)}(\boldsymbol{g}') \,, \tag{2}$$

where $\phi^{-1}(p) := \{\boldsymbol{g}' \in \mathcal{G} \mid \phi(\boldsymbol{g}') = p\}$ is called the *neutral set* of $p \in \mathcal{P}$ [25]. Thus, the genotype space $\mathcal{G}$ together with the genotype-phenotype mapping ϕ, the variation operators, and external parameters $\boldsymbol{\theta}$ can be regarded as a parameterization of the exploration distribution on the search space $\mathcal{P}$. We refer to [31] for more details on this way of formalizing evolutionary exploration.

Also algorithms recently developed in the field of evolutionary computing can be captured by this point of view: Mühlenbein et al. [16] and Pelikan et al. [19] parameterize the exploration density by means of a Bayesian dependency network in order to introduce correlations in the exploration distribution. The CMA evolution strategy proposed by Hansen and Ostermeier [11] adapts a covariance matrix that describes the dependencies between real-valued variables in the exploration density. In the remainder of this paper, we focus on a different approach that might be considered biologically more plausible, namely to utilize an appropriate genotype-phenotype mapping to parameterize the exploration density $P_{\mathcal{P}}$. It does not explicitly encode correlations but models such interactions indirectly via the genotype-phenotype mapping. What we will ask for in the next section is the way it allows for self-adaptation of the exploration strategy.

III. Self-adaptation

A. Introduction

The ability of an evolutionary algorithm to adapt its search strategy during the optimization process is a key concept in evolutionary computation, see the overviews [1, 29, 8]. Online adaptation of strategy parameters is important, because the best setting of an EA is usually not known *a priori* for a given task and a constant search strategy is usually not optimal during the evolutionary process. One way to adapt the search strategy online is self-adaptation, see [4] for an overview. This method can be described as follows [8]: "The idea of the evolution of evolution can be used to implement the self-adaptation of parameters. Here the parameters to be adapted are encoded into the chromosomes and undergo mutation and recombination. The better values of these encoded parameters lead to better individuals, which in turn are more likely to survive and produce offspring and hence propagate these better parameter values." In other words, each individual not only represents a candidate solution for the problem at hand, but also certain strategy parameters that are subject to the same selection process—they hitchhike with the object parameters.

Self-adaptation is used in all main paradigms of evolutionary computation. The most prominent examples stem from evolution strategies, where it is used to adapt the covariance matrix of the mutation operator, see Sec. IV-B. Self-adaptation is employed in evolutionary programming for the same purpose, but also in the original framework of finite-state machines [10]. In genetic algorithms, the concept of self-adaptation has been used to adapt mutation probabilities [3] and crossover operators [23]. Self-adaptive crossover operators have also been investigated in genetic programming [2]. In the following, we will propose an alternative definition of self-adaptation, which includes these approaches as special cases.

B. Neutrality and self-adaptation

Following the formalism we developed in the previous section, a variation of the exploration density $P_{\mathcal{P}}^{(t)}$, corresponding to Step 3 in the global random search algorithm, occurs if

1. the parent population $(\tilde{\boldsymbol{g}}_1^{(t)}, \dots, \tilde{\boldsymbol{g}}_\mu^{(t)})$ varies, or
2. the external strategy parameters $\boldsymbol{\theta}^{(t)}$ vary, or
3. the genotype-phenotype mapping ϕ varies.

From a biological point of view, one might associate environmental conditions (like the temperature, etc.) with external parameters that vary the mutation probabilities or the genotype-phenotype mapping. In most cases one would not consider them as subject to evolution.[2] Sometimes mechanisms that adapt the exploration distribution by genotype variations are regarded as examples of adaptive genotype-phenotype mappings. To our minds, this is a misleading point of view. For example, t-RNA determines a part of the genotype-phenotype mapping and it is itself encoded in the genome. However, the genotype-phenotype mapping should be understood as a whole—mapping all the genotype (including the parts that code for the t-RNA) to the phenotype, such that it becomes inconsistent to speak of genotypic information parameterizing the genotype-phenotype mapping.

The same arguments also apply in the context of evolutionary computation and thus we consider only option 1 as a possibility to vary the exploration distribution in a way that is itself subject to evolution in the sense of section III-A. However, if the genotype-phenotype mapping is injective, every variation of genotypes alters phenotypes and bears the risk of a fitness loss. Hence, we conclude:

In the absence of external control, only neutral genetic variations can allow a self-adaptation of the exploration distribution without changing the phenotype, i.e., without the risk of loosing fitness.

A *neutral genetic variation* means that parent and offspring have the same phenotype. For instance, consider two genotypes $\tilde{\boldsymbol{g}}_1$, $\tilde{\boldsymbol{g}}_2$ in a neutral set. Neglect crossover and assume that the probability for an offspring $\boldsymbol{g}$ of a single parent $\tilde{\boldsymbol{g}}_i$ is given by $P_{\mathcal{G}}(\boldsymbol{g} \mid \tilde{\boldsymbol{g}}_i; \boldsymbol{\theta})$. The two genotypes may induce two arbitrarily different exploration distributions "around" the same phenotype $p = \phi(\tilde{\boldsymbol{g}}_1) = \phi(\tilde{\boldsymbol{g}}_2)$, see Fig. 1. Transitions between these genotypes allow for switching the exploration strategy. In general, the variety of exploration densities that can be explored in a neutral set $\phi^{-1}(p)$ is given by

$$\left\{ P_{\mathcal{P}}(p \mid \tilde{\boldsymbol{g}}_i; \boldsymbol{\theta}) \;\middle|\; \tilde{\boldsymbol{g}}_i \in \phi^{-1}(p) \right\} . \tag{3}$$

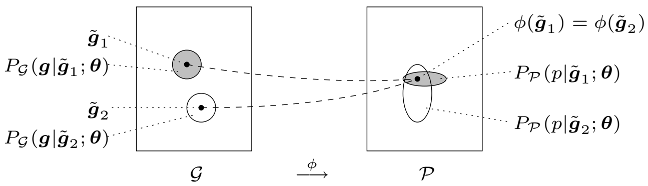
$\tilde{\boldsymbol{g}}_1$
$P_{\mathcal{G}}(\boldsymbol{g}|\tilde{\boldsymbol{g}}_1;\boldsymbol{\theta})$
$\tilde{\boldsymbol{g}}_2$
$P_{\mathcal{G}}(\boldsymbol{g}|\tilde{\boldsymbol{g}}_2;\boldsymbol{\theta})$
$\phi(\tilde{\boldsymbol{g}}_1) = \phi(\tilde{\boldsymbol{g}}_2)$
$P_{\mathcal{P}}(p|\tilde{\boldsymbol{g}}_1;\boldsymbol{\theta})$
$P_{\mathcal{P}}(p|\tilde{\boldsymbol{g}}_2;\boldsymbol{\theta})$
$\mathcal{G}$
$\xrightarrow{\phi}$
$\mathcal{P}$

Fig. 1. Two different points $\tilde{\boldsymbol{g}}_1, \tilde{\boldsymbol{g}}_2$ in $\mathcal{G}$ are mapped onto the same point in $\mathcal{P}$. The elliptic ranges around the points illustrate the exploration distributions by suggesting the range of probable mutants. Thus, the two points $\tilde{\boldsymbol{g}}_1, \tilde{\boldsymbol{g}}_2$ belong to one neutral set but represent two different exploration strategies on the search space $\mathcal{P}$.

C. Discussion

It is widely accepted that changing the genotypes without significantly changing the phenotypes in the population is a key search strategy in natural evolution [12, 14, 24]. We emphasize that under the stated assumptions, neutrality is even a necessity for self-adaptation of the search strategy. Thus, we propose to *define self-adaptation as the use of neutrality in order to vary the exploration strategy.*

Existing approaches to self-adaptation developed in the realm of evolutionary computation can be embedded in this point of view; only the *style* in which the notion of self-adaptation was originally introduced is different. Usually one refers to "strategy parameters" that typically control the mutation operators (i.e., the mapping $\tilde{\boldsymbol{g}}_i \mapsto P_{\mathcal{G}}(\boldsymbol{g} \mid \tilde{\boldsymbol{g}}_i; \theta)$), in contrast to "object parameters" describing fitness relevant traits. In the case of *self*-adaptation these strategy parameters are considered as parts of the genotype. In this view, strategy parameters are neutral, i.e., altering them does not change the phenotype. In turn, we regard neutral genotypic variables as potential strategy parameters. There still is a slight difference: In case of standard self-adaptation, altering a strategy parameter is phenotypically neutral and has no other implication than a change of the exploration strategy. In contrast, two or more genetic variables may be phenotypically non-neutral but a specific simultaneous mutation of them might be neutral and induce a change of the exploration density. Hence, a neutral set is a general concept for self-adaptation, which is not bound to the idea of single loci being responsible for the search strategy. A good example is that also topological properties of the search space (i.e., neighborhood relations/the set of most probable mutations of a phenotype) may vary along such a neutral set if the genotype-phenotype mapping is chosen similar to grammars [31]—which seems hard to realize by explicit strategy parameters.

We believe that allowing for a self-adaptive exploration strategy is the main benefit of neutral encodings, as stated in [27]: "For redundancy to be of use it must allow for mutations that do not change the current phenotype, thus maintaining fitness, and which allow for moves to

[2] Counter-examples that go far beyond the scope of our formalism are, for instance, the embryonic environment (uterus) and the inherited ovum, or individuals whose behavior have an influence on mutations (e.g., sexual behavior influencing crossover) or on the embryonic development (e.g., a mother taking drugs). All of these influences might be considered as subject to evolution. In particular the interpretation of an ovum is a critical issue. Should one regard it as part of the genotype or as an inherited "parameter" of the genotype-phenotype mapping?

areas where new phenotypes are accessible". Changing the exploration distribution corresponds to the ability of reaching new phenotypes.[3]

In this view, neutrality is not necessarily *redundant*: Different points in a neutral set can encode different exploration strategies and thus different information; a genotype encodes not only the information on the phenotype but also information on further explorations. One might state it like this: Although a non-injective genotype-phenotype mapping can be called redundant, the corresponding mapping from genotype to exploration distribution may in general be non-redundant.

IV. Examples

A. Codon Bias in natural evolution

An intriguing study of the interrelationship between neutrality and self-adaptation in nature is the one by Stephens and Waelbroeck [30]. They empirically analyze the codon bias and its effect in RNA sequences of the HI virus. In biology, several nucleotide triplets eventually encode the same amino acid. For example, there are 9 triplets that are mapped to the amino acid Arginine. If Arginine is encoded by the triplet CGA, then the chance that a single point mutation within the triplet is neutral (does not chance the encoded amino acid) is 4/9. In contrast, if it is encoded by AGA, then this *neutral degree* is 2/9. Now, codon bias means that, although there exist several codons that code for the same amino acid (which form a neutral set), HIV sequences exhibit a preference on which codon is used to code for a specific amino acid. More precisely, at some places of the sequence codons are preferred that are "in the center of this neutral set" (with high neutral degree) and at other places codons are biased to be "on the edge of this neutral set" (with low neutral degree). It is clear that these two cases induce different exploration densities; the prior case means low mutability whereas the latter means high mutability. Stephens and Waelbroeck go further by giving an explanation for these two (marginal) exploration strategies: Loci with low mutability cause "more resistance to the potentially destructive effect of mutation", whereas loci with high mutability might induce a "change in a neutralization epitope which has come to be recognized by the [host's] immune system" [30].

B. Self-adaptation in evolution strategies

In this section, we describe a rather simple example of self-adaptation in evolution strategies.[4] The candidate solutions for the problem at hand are represented by real-valued vectors $\boldsymbol{x} \in \mathcal{P} = \mathcal{G} \subseteq \mathbb{R}^n$. These real-valued object parameters are mutated by adding a realization of a normally distributed random vector with zero mean [21, 26]. The symmetric $n \times n$ covariance matrix of this random vector is subject to self-adaptation. To describe any possible covariance matrix, $n(n+1)/2$ strategy parameters are needed. However, often a reduced number of parameters is used. In the following, let the covariance matrix be given by $(\boldsymbol{\sigma} I)^2$, where $I = \text{diag}(1, \dots, 1)$ is the identity matrix and $\boldsymbol{\sigma} \in \mathbb{R}^n$ corresponds to n strategy parameters that describe the standard deviations along the coordinate axes.

Let μ denote the size of the parent population. Each generation, λ offspring $\boldsymbol{g}_i^{(t)}$, $i = 1, \dots, \lambda$ are generated. (μ, λ)-selection is used, i.e., the μ best individuals of the offspring form the new parent population. An individual $\boldsymbol{g}_i^{(t)} \in \mathbb{R}^{2n}$ can be divided into two parts, $\boldsymbol{g}_i^{(t)} = (\boldsymbol{x}_i^{(t)}, \boldsymbol{\sigma}_i^{(t)})$, the object variables $\boldsymbol{x}_i^{(t)} \in \mathbb{R}^n$ and the strategy parameters $\boldsymbol{\sigma}_i^{(t)} \in \mathbb{R}^n$.

For each offspring $\boldsymbol{g}_i^{(t)}$, two indices $a, b \in \{1, \dots, \mu\}$ are selected randomly, where a determines the parent and b its mating partner. For each component $k = 1, \dots, n$ of the offspring, the new standard deviation $\boldsymbol{\sigma}_i^{(t)} = (\sigma_{i,1}^{(t)}, \dots, \sigma_{i,n}^{(t)})$ is computed as

$$\sigma_{i,k}^{(t)} = \frac{1}{2}\left[\tilde{\sigma}_{a,k}^{(t)} + \tilde{\sigma}_{b,k}^{(t)}\right] \cdot \exp\left(\tau' \cdot \xi_i^{(t)} + \tau \cdot \xi_{i,k}^{(t)}\right) \quad . \tag{4}$$

Here, $\xi_{i,k}^{(t)} \sim \mathcal{N}(0,1)$ is a realization of a normally distributed random variable with zero mean and variance one that is sampled anew for each component i for each individual, whereas $\xi_i^{(t)} \sim \mathcal{N}(0,1)$ is sampled once per individual and is the same for each component. The log-normal distribution ensures that the standard deviations stay positive. For the mutation strengths $\tau \propto 1/\sqrt{2\sqrt{n}}$ and $\tau' \propto 1/\sqrt{2n}$ are recommended [26, 4]. It has been shown that recombination of the strategy parameters is beneficial (e.g., [15]). Equation (4) realizes an intermediate recombination of the strategy parameters.

Thereafter the objective parameters are altered using the new strategy parameters:

$$x_{i,k}^{(t)} = \tilde{x}_{a,k}^{(t)} + \sigma_{i,k}^{(t)}\, z_{i,k}^{(t)}, \tag{5}$$

where $z_{i,k}^{(t)} \sim \mathcal{N}(0,1)$. For simplicity, we do not employ recombination of the objective parameters. An example of this strategy compared to algorithms without self-adaptation is shown in Fig. 3. The adaptive strategy performs considerably better than the other methods, as known from many theoretical results and empirical studies. However, we would like to underline that the self-adaptive EA uses a highly redundant encoding instead of a one-to-one genotype-phenotype mapping—the genotype space has twice the dimensionality of the phenotype space. The neutrality does not alter the distribution of

[3] However, we think that there might be at least one additional case where neutrality can be useful, namely when it introduces—*by chance*—a bias in the search space, such that desired phenotypes are (in global average) represented more often than other ones [13].

[4] More sophisticated and efficient algorithms exist for dealing with real-world optimization tasks, e.g., the derandomized adaptation as proposed in [11, 18].

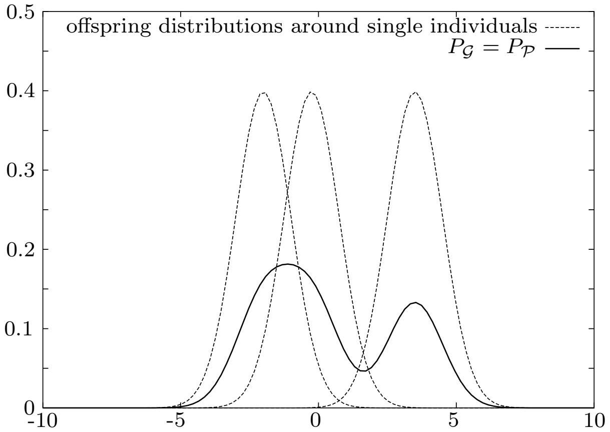

Fig. 2. An example for parameterizing the exploration distribution in evolution strategies: Consider the search space $\mathcal{P} = \mathbb{R}$, $\mu = 3$ parent individuals, and no recombination. The parents represent the object parameters -2, -0.25, and 3.5. The offspring is generated in the following manner: First, one randomly chosen parent is reproduced. Second, the reproduced individual is mutated by adding a realization of a normally distributed random number with variance one and expectation zero. Hence, the resulting search distribution (the joint generating distribution [22]) is a multimodal mixture of Gaussians.

the fitness values, it just allows the search strategy to adapt.

V. Conclusion

Neutrality is a necessity for self-adaptation. Actually, the design of neutral encodings to improve the efficiency of evolutionary algorithms is a well-established approach: strategy parameters are an example of neutrality. Hence, there already exists clear evidence for the benefit of neutrality.

The notion of neutrality provides a unifying formalism for embedding approaches to self-adaptation in evolutionary computation. But it also inspires new approaches that are not bound to the idea of only single genes being responsible for the exploration strategy. Generally, any local properties of the phenotypic search space—metric or topological—may vary along a neutral set. An example are grammar-based genotype-phenotype mappings, for which different points in a neutral set represent completely different topologies (and thus exploration strategies) on the phenotype space.

On the other hand, the benefits of neutrality in natural evolution can be better understood when the simple—but concrete and well-established—paradigms of self-adaptation and neutrality in evolutionary computation are taken into account. As we exemplified with the second example, these computational approaches may serve as a model to investigate the relation between evolutionary progress, neutrality, and self-adaptation.

Finally, let us recall that neutrality is not necessarily equivalent to redundancy: in general, a genotype encodes

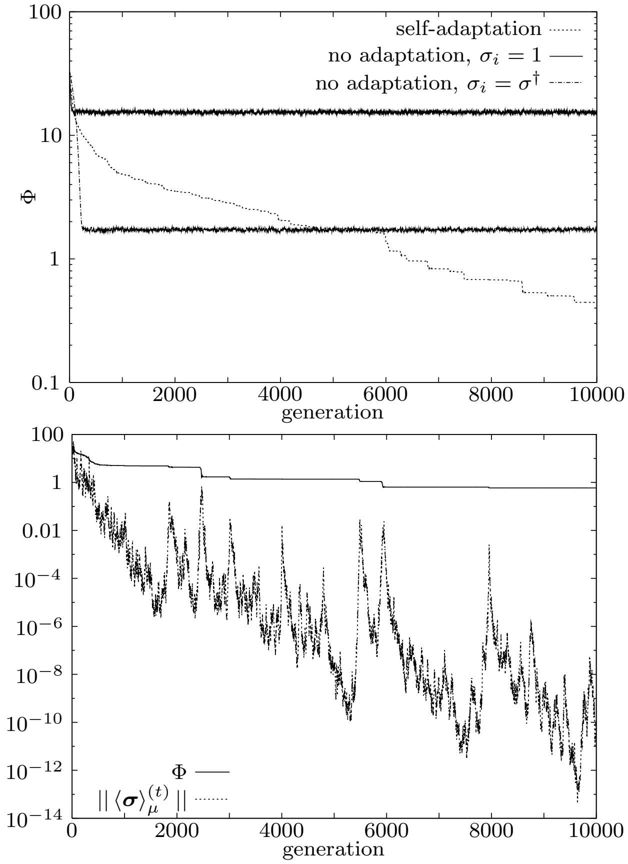

Fig. 3. As an example, we compare evolution strategies as described in Sec. IV-B with and without self-adaptation on a simple test problem: a 100-dimensional sphere model, $\Phi(\boldsymbol{x}) = ||\boldsymbol{x}||$. First, the EA is run without self-adaptation, each σ_i is set to one. After that, we employ self-adaptation as described, where the σ_i are also initialized to one. Based on these results, we determine the average standard deviation $\sigma^\dagger$ (averaged over all parents, all generations, and all trials). Then we repeat the experiments without self-adaptation but fix each $\sigma_i = \sigma^\dagger$. The fitness trajectories of the best individual averaged over 50 trials are shown in the upper plot ($\sigma^\dagger = 0.0125$). In case of the self-adaptive EA the fitness curves show steps. The lower plot shows a typical single trial and the corresponding average of the standard deviations $||\,\langle \boldsymbol{\sigma} \rangle_\mu^{(t)}\,|| = \left(\sum_{j=1}^{n} \left(\frac{1}{\mu} \sum_{i=1}^{\mu} \sigma_{i,j} \right)^2 \right)^{1/2}$. It becomes obvious that each step in the fitness trajectory corresponds to a change in the search strategy. In general, the mutation strength decreases, but sometimes a larger step size takes over the population and leads to a high fitness gain. In the continuum, the probability that an offspring has exactly the same phenotype, i.e., represents the same object parameters, is not measurable. Further, even small changes in the genotype are relevant for selection due to the use of a rank-based selection scheme. Hence, *drifting* along a *neutral network*, i.e., traversing the search space by a sequence of (*neutral*) mutations that do not alter the phenotype, does not occur.

not only the information on the phenotype but also information on further explorations.

Acknowledgments

The authors acknowledge support by the DFG under grant Solesys SE 251/41-1.

References

[1] P. J. Angeline. Adaptive and self-adaptive evolutionary computations. In M. Palaniswami, Y. Attikiouzel, R. Marks, D. B. Fogel, and T. Fukuda, editors, *Computational Intelligence: A Dynamic Systems Perspective*, pages 152–163. IEEE Press, 1995.

[2] P. J. Angeline. Two self-adaptive crossover operations for genetic programming. In P. Angeline and K. Kinnear, editors, *Advances in Genetic Programming*, volume 2, Cambridge, MA, 1996. MIT Press.

[3] T. Bäck. Self–adaptation in genetic algorithms. In F. J. Varela and P. Bourgine, editors, *Towards a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, pages 263–271. MIT Press, 1992.

[4] T. Bäck. An overview of parameter control methods by self-adaptation in evolutionary algorithms. *Fundamenta Informaticae*, 35(1-4):51–66, 1998.

[5] L. Barnett. Ruggedness and neutrality –The NKp family of fitness landscapes. In C. Adami, R. K. Belew, H. Kitano, and C. E. Taylor, editors, *Alive VI: Sixth International Conference on Articial Life*, pages 18–27. MIT Press, 1998.

[6] M. Conrad. The geometry of evolution. *Biosystems*, 24:61–81, 1990.

[7] M. Ebner, P. Langguth, J. Albert, M. Shackleton, and R. Shipman. On neutral networks and evolvability. In *Proceedings of the 2001 Congress on Evolutionary Computation*, pages 1–8. IEEE Press, 2001.

[8] A. E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 3(2):124–141, 1999.

[9] D. B. Fogel. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence.* IEEE Press, 1995.

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

[11] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001.

[12] M. A. Huynen. Exploring phenotype space through neutral evolution. *Journal of Molecular Evolution*, 43:165–169, 1996.

[13] C. Igel and P. Stagge. Effects of phenotypic redundancy in structure optimization. *IEEE Transactions on Evolutionary Computation.* In press.

[14] M. Kimura. Evolutionary rate at the molecular level. *Nature*, 217:624–626, 1968.

[15] F. Kursawe. *Grundlegende empirische Untersuchungen der Parameter von Evolutionsstrategien — Metastrategien.* Dissertation, Fachbereich Informatik, Universität Dortmund, 1999.

[16] H. Mühlenbein, T. Mahnig, and A. O. Rodriguez. Schemata, distributions and graphical models in evolutionary optimization. *Journal of Heuristics*, 5(2):215–247, 1999.

[17] M. E. J. Newman and R. Engelhardt. Effects of neutral selection on the evolution of molecular species. *Proceedings of the Royal Society of London, Series B: Biological Sciences*, 265(1403):1333–1338, 1998.

[18] A. Ostermeier, A. Gawelczyk, and N. Hansen. A derandomized approach to self-adaptation of evolution strategies. *Evolutionary Computation*, 2(4):369–380, 1995.

[19] M. Pelikan, D. E. Goldberg, and E. Cantú-Paz. Linkage problem, distribution estimation, and bayesian networks. *Evolutionary Computation*, 9:311–340, 2000.

[20] N. J. Radcliffe. Equivalence class analysis of genetic algorithms. *Complex Systems*, 5:183–205, 1991.

[21] I. Rechenberg. *Evolutionsstrategie '94.* Werkstatt Bionik und Evolutionstechnik. Frommann-Holzboog, Stuttgart, 1994.

[22] G. Rudolph. Massively parallel simulated annealing and its relation to evolutionary algorithms. *Evolutionary Computation*, 1(4):361–382, 1994.

[23] J. D. Schaffer and A. Morishima. An adaptive crossover distribution mechanism for genetic algorithms. In J. J. Grefenstette, editor, *Proceedings of the Second International Conference on Genetic Algorithms and Their Applications (ICGA'87)*, pages 36–40. Lawrence Erlbaum Associates, 1987.

[24] P. Schuster. Molecular insights into evolution of phenotypes. In J. P. Crutchfield and P. Schuster, editors, *Evolutionary Dynamics—Exploring the Interplay of Accident, Selection, Neutrality and Function*, Santa Fe Institute Series in the Science of Complexity. Oxford University Press. To appear.

[25] P. Schuster. Landscapes and molecular evolution. *Physica D*, 107:331–363, 1996.

[26] H.-P. Schwefel. *Evolution and Optimum Seeking.* Sixth-Generation Computer Technology Series. John Wiley & Sons, New York, 1995.

[27] M. A. Shackleton, R. Shipman, and M. Ebner. An investigation of redundant genotype-phenotype mappings and their role in evolutionary search. In A. Zalzala, C. Fonseca, J.-H. Kim, and A. Smith, editors, *Proceedings of the International Congress on Evolutionary Computation (CEC 2000)*, pages 493–500. IEEE Press, 2000.

[28] R. Shipman. Genetic redundancy: Desirable or problematic for evolutionary adaptation. In M. A. Bedau, S. Rasmussen, J. S. McCaskill, and N. H. Packard, editors, *Proceedings of the 4th International Conference on Artificial Neural Networks and Genetic Algorithms (ICANNGA)*, pages 337–344. Springer-Verlag, 1999.

[29] J. E. Smith and T. C. Fogarty. Operator and parameter adaptation in genetic algorithms. *Soft Computing*, 1(2):81–87, 1997.

[30] C. R. Stephens and H. Waelbroeck. Codon bias and mutability in HIV sequences. *Journal of Molecular Evolution*, 48:390–397, 1999.

[31] M. Toussaint. Self-adaptive exploration in evolutionary search. Technical Report IRINI 2001-05, Institut für Neuroinformatik, Lehrstuhl für Theoretische Biologie, Ruhr-Universität Bochum, 44780 Bochum, Germany, 2001.

[32] K. Weicker and N. Weicker. Burden and benefits of redundancy. In *Proceedings ot the Sixth Workshop on Foundations of Genetic Algorithms*, 2000.

[33] A. A. Zhigljavsky. *Theory of global random search.* Kluwer Academic Publishers, 1991.