Evolution Strategy with Neighborhood Attraction Using a Neural Gas Approach

Jutta Huhse, Thomas Villmann*, Peter Merz and Andreas Zell
[huhse|pmerz|zell]@informatik.uni-tuebingen.de,
*villmann@informatik.uni-leipzig.de

University of Tübingen
Inst. of Computer Science
D–72076 Tübingen, Sand 1, Germany
*University of Leipzig
Clinic for Psychotherapy and Psychosomatic Medicine
D–04107 Leipzig, Karl–Tauchnitz–Str.25, Germany

Abstract. In evolution strategies with neighborhood attraction (EN) the concepts of neighborhood cooperativeness and learning rules known from neural maps are transferred onto the individuals of evolution strategies. A previous approach, which utilized a neighborhood relationship adapted from self-organizing maps (SOM), appeared to perform as well as or even better than comparable conventional evolution strategies on a variety of common test functions. In this contribution, an EN with a new neighborhood relationship and learning rule based on the idea of neural gas is introduced. Its performance is compared to the SOM-like approach, using the same test functions. It is shown that the neural gas approach is considerably faster in finding the optimum than the SOM approach, although the latter seems to be more robust for multi-modal problems.

1 Introduction

Evolutionary algorithms (EAs) are stochastic search algorithms inspired by principles of biological evolution working simultaneously on a large number of *potential* problem solutions. Thereby, these solutions are judged by an external cost function, called fitness.

The power of *neural networks* (NNs) arises from complex, dynamic connections between quite simple calculation units (called neurons) working in a massively parallel manner. These features enable neural networks to be both robust and adaptive.

The combination of neural networks and EAs offers new possibilities to increase the power of adaptive approaches. Two main directions can be identified: the first one is to apply EAs to optimize a neural network whereas the second one deals with the transfer of methods from neural networks onto the framework of EAs. For the first type of combining EAs and NNs many examples can be found in the literature. For the second type however, as already stated in [1], there are only a few approaches of incorporating aspects of artificial neural networks into evolutionary algorithms.

Taking this into account, in this paper we focus on such hybrid systems which combine EAs with *neural maps* as a special type of neural networks. Examples are Kohonen's widely-used *self-organizing map* (SOM) [2] and the *topology representing network* (TRN) [3]. The main paradigm of neural map learning distinguishing it from other network types is the incorporation of neighborhood oriented learning, i. e. neighboring neurons try to adapt in a similar fashion.

Based on previous work, where an *E*volution strategy with *N*eighborhood attraction (EN) on the basis of Kohonen's SOM was introduced and compared to conventional evolution strategies [4], here we present a new neighborhood and attraction scheme for this EN, based on neural gas (NG) [5].

First, an introduction to neural maps is given in section 2. Then, based on the description of the SOM inspired EN (subsection 3.1) an EN with a new, neural gas like neighborhood is explained in subsection 3.2. This new approach is analyzed using several well-known test functions (section 4) and the results are given in section 5.

2 Neural Maps

Neural maps as tools for (unsupervised) topographic vector quantization project data from a (possibly high-dimensional) input space $\mathcal{V} \subseteq I\!R^{D_\mathcal{V}}$ onto a position in some output space (neuron lattice) $\mathcal{A}$, such that a continuous change of a parameter of the input data should lead to a continuous change of the position of a localized excitation in the grid. This property is called *neighborhood preserving* (topographic) mapping. In the most widely used neural map algorithm, the *self-organizing map* (SOM) [2], the N neurons are arranged *a priori* on a fixed grid, i. e. the $\mathbf{r_i} \in \mathcal{A}$ denote the positions of the neurons in the grid $\mathcal{A}$. Usually the grid is chosen as a hypercube, but other structured arrangements are also admissible.

Other algorithms, such as the TRN, based on the neural gas algorithm (NG) [5, 6], do not specify the topological relations among neurons in $\mathcal{A}$ in advance, but dynamically adapt the neighborhood structure according to the data distribution during learning.

Associated with each neuron is a weight vector (or reference vector) $\mathbf{w_r} \in I\!R^{D_\mathcal{V}}$. The mapping is realized by $\Psi_{\mathcal{V}\to\mathcal{A}} : \mathbf{v} \mapsto \mathbf{s}(\mathbf{v}) = \arg\min_{\mathbf{r}\in\mathcal{A}} \parallel \mathbf{v} - \mathbf{w_r} \parallel$, where $\parallel \mathbf{v} - \mathbf{w_r} \parallel$ is the Euclidean distance between a given input signal $\mathbf{v}$ and the weight vector $\mathbf{w_r}$. The neuron $\mathbf{s}$ is referred to as the *winner* or *best–matching unit.* During learning the weight vectors are adapted according to

$$\Delta\mathbf{w_r} = \epsilon \cdot h_\sigma(\mathbf{r}, \mathbf{s}, \mathbf{v})(\mathbf{v} - \mathbf{w_r}). \tag{1}$$

Thereby $h_\sigma(\mathbf{r}, \mathbf{s}, \mathbf{v})$ is the neighborhood function usually chosen to be of Gaussian shape

$$h_\sigma(\mathbf{r}, \mathbf{s}, \mathbf{v}) = \exp\left(-\frac{(d(\mathbf{r}, \mathbf{s}(\mathbf{v})))^2}{2\sigma^2}\right) \tag{2}$$

where $d(\mathbf{r}, \mathbf{s}(\mathbf{v}))$ is a certain distance measure defined on the set $\mathcal{A}$. For SOM, $d(\mathbf{r}, \mathbf{s}(\mathbf{v}))$ is evaluated in the grid $\mathcal{A}$, whereas for TRN, it is evaluated in the input (problem) space $\mathcal{V}$. With $h_\sigma(\mathbf{r}, \mathbf{s}, \mathbf{v})$, a neighborhood oriented cooperativeness for learning is installed which is triggered by the neighborhood range σ of the neighborhood function. During the learning process usually σ as well as the learning rate ϵ decrease to small values referred to as remaining cooperativeness σ_{final} and learnability ϵ_{final} of the network. In the beginning of the adaptation the neighborhood range and the learning rate are high which corresponds to a *rough but fast learning* whereas in the convergence phase the rough adaptation is replaced by a *fine tuning*. The first phase is indicated by a high cooperativeness between the neurons which realizes a high information transfer. In the fine tuning phase the neurons act more or less separately with an only small (nearest neighbor) but not vanishing communication.

3 Evolution Strategies with Neighborhood Attraction

As mentioned above there are not many approaches that incorporate concepts of neural networks into EA to improve EA-performance [7, 1]. This fact certainly holds also for the special combination of neural maps and ES. Here, we concentrate on evolution strategies with neighborhood attraction (EN) [4]. The idea behind EN is to transfer the neighborhood and the learning rule known from neural maps onto the originally unstructured individuals of an ES to concentrate the individuals around the optimum.

3.1 SOM-Neighborhood attraction EN

The SOM-Neighborhood attraction EN (SOM-EN) introduced by Huhse and Zell [4, 8] works as follows:

The *neighborhood* between the μ parent individuals is constituted by arranging them on an orthogonal, elastic grid. As known from the SOM, each individual can be identified by its fixed grid position. Two individuals a_i and a_j are grid neighbors if they are directly connected on the grid; i.e. $h_\sigma = 1$ (see equ. 2). If no direct grid connection exists, they are not neighbors ($h_\sigma = 0$).

In contrast to conventional ES and SOM, the initial values of the object variables of the EN individuals are not assigned randomly. Rather, the problem space is divided into equally sized hypercubes, each of them corresponding to one grid position. The object variables of the associated individual are initialized with equally distributed random values within the ranges of its hypercube.

As is customary in ES, the EN individuals are evaluated using a certain fitness function.

For the evolutionary *variation* an EN-specific operator – the neighborhood attraction – is introduced which manipulates the EN individuals according to one learning step in a SOM (cf. 1). Each parent individual a_P is successively selected. Among all its (direct grid) neighbors the best neighbor a_{Nb} is determined. If the fitness value of the neighbor is better than the fitness value of the parent, the

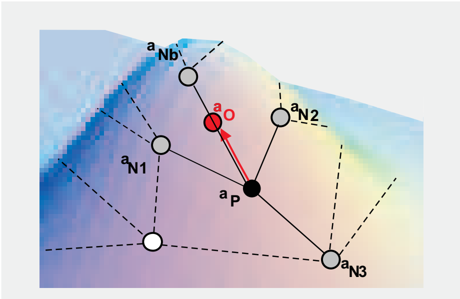

Fig. 1. Neighborhood attraction in SOM-EN

EN-specific neighborhood attraction is applied: An offspring individual a_O is produced by attracting the parent individual a_P towards its best neighbor a_{Nb} (see Fig. 1).

In detail, the variation is realized on the level of the object variables by equation 3, with the neighborhood attraction operator (equ. 4). The neighborhood relations are retained unchanged.

$$\boldsymbol{x}_O = \boldsymbol{x}_P + \Delta\boldsymbol{x}_P \tag{3}$$

$$\Delta\boldsymbol{x}_{\boldsymbol{P}} = \delta \cdot (\boldsymbol{x}_{Nb} - \boldsymbol{x}_P) \tag{4}$$

Thereby, the *neighborhood attraction factor* δ defines the strength of the attraction along the difference vector. One parent individual can produce n_O offspring by varying δ.

If the fitness value of the parent is considered better than those of all its neighbors a_{Nj} a "simple conventional" mutation (referred to as *ES-mutation* here) is performed. n_O offspring are generated according to equ. 5.

$$\begin{aligned} \boldsymbol{z} &= \boldsymbol{\mathcal{N}}(0,1) \\ s_{\mathit{eff}} &= \tfrac{1}{n} d_{min} \\ \Delta\boldsymbol{x}_P &= s_{\mathit{eff}} \cdot \boldsymbol{z} \end{aligned} \tag{5}$$

$\boldsymbol{z}$ is a vector of normally distributed random numbers. The effective step size s_{eff} is determined by $\frac{1}{n}$, the reciprocal number of object variables and by the distance $d_{min} = \min(||\boldsymbol{x}_P - \boldsymbol{x}_{Nj}||)$ to the nearest neighbor. Thus, a mutation which jumps over a neighbor and an entanglement of the grid becomes less likely. During the contraction of the grid the effective step sizes decrease due to the influence of d_{min}.

Deviating from the in ES commonly used $(\mu + \lambda)$- or (μ, λ)-selection scheme [9] – where the μ individuals which go to the next generation are selected from all μ parent *plus* all λ offspring individuals or *only* from all λ offspring, resp. – a *parental* selection scheme is used here, which resembles brood selection known from genetic programming [10]. Selection is performed for every parent a_P and its n_O offspring separately. Each parent contributes exactly one individual to the next generation. If the n_O offspring have been produced by neighborhood attraction, the parent is always replaced by its best offspring, even if the offspring's fitness is worse (similar to (μ, λ)-selection). Allowing such temporary fitness decrease, an individual can pass through a local optimum towards better fitness values. If the n_O offspring have been produced by ES-mutation, the best of the parent and its offspring is selected (similar to $(\mu + \lambda)$-selection). Using parental selection, one the one hand, the grid neighborhood is not disrupted by the selection process, and, on the other hand, diversity of the population is better maintained.

3.2 Neural Gas Attraction EN

Now, we propose a new neighborhood and attraction scheme, the neural gas attraction EN (NG-EN). According to the idea in the NG mentioned above, we define the topological neighborhood h_σ for the individuals depending on their distances $d(a_i, a_j)$ in the problem space. The neighborhood radius for each individual a_P is determined by its average (Euclidean) distance $\hat{d_E} = \frac{1}{n}\sum_i (\| \boldsymbol{x_i} - \boldsymbol{x_P} \|)$ to all other individuals. All individuals a_{Nj} inside this radius are considered to be neighbors of a_P.

Like in SOM-EN, *variation* of the individuals is performed on each parent individual a_P separately: All its neighbor individuals a_{Nj} are ranked according to their increasing distances to a_P, as it is known from NG. The neighborhood h_σ decreases with the neighbor's rank and with time (equ. 7). Then a_P places λ offspring in different directions, offspring a_{Oi} in direction to the neighbor a_{Ni} with rank i.

In analogy to the neural gas algorithm [5, 6, 3], the attraction rules from equations (1), (2) and (4) are rewritten as

$$\Delta \boldsymbol{x_P}(i) = \epsilon(t) \cdot h_\sigma(i,t) \cdot (\boldsymbol{x_{Ni}} - \boldsymbol{x_P}) \tag{6}$$

$$h_\sigma(i,t) = \exp\left(\frac{-\text{rank}(i)}{\sigma(t)}\right) \tag{7}$$

$$\sigma(t) = \sigma_i \cdot (\sigma_f/\sigma_i)^{(t/t_{max})}$$

$$\epsilon(t) = \epsilon_i \cdot (\epsilon_f/\epsilon_i)^{(t/t_{max})}$$

with

$$\delta(i,t) = \epsilon(t) \cdot h_\sigma(i,t) \quad \text{and} \quad \delta \in [\delta_{min}, \delta_{max}].$$

Fig. 2 shows the neighborhood attraction δ decreasing with time and rank. Two applicable parameter settings were chosen: While $\sigma_i = 5$, $\sigma_f = 0.05$ and $t_{max} = 30000$ are constant, ϵ_i and ϵ_f were set to either $\epsilon_i = 0.0075$ and $\epsilon_f = 0.00375$ (left) or $\epsilon_i = 0.02$ and $\epsilon_f = 0.01$ (right). The initial value for the resulting δ is in compliance with suggestions made for the δ used in SOM-EN [8].

As in the SOM-EN, this neural gas neighborhood attraction is performed only if a_P is not better than its neighbors. Otherwise the ES-mutation known from (5) is applied.

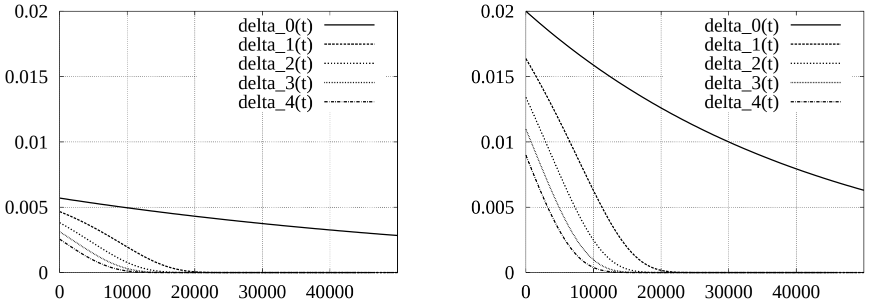

Fig. 2. Neighborhood attraction factor δ_i decreasing with time t and rank i. **Left:** for the neural gas parameter settings ($\sigma_i = 5$, $\sigma_f = 0.05$, $\epsilon_i = 0.0075$, $\epsilon_f = 0.00375$), **right:** ($\sigma_i = 5$, $\sigma_f = 0.05$, $\epsilon_i = 0.02$, $\epsilon_f = 0.01$).

4 Test Functions

To investigate the new neighborhood attraction scheme we conducted thorough test series on an exhaustive test set of well-known optimization tasks also used in e. g. [11–13] and in former investigations on the SOM-EN (see Table 1). The dimensions were set to 20, as long as there were no other ranges predefined. In those cases the highest acceptable dimension was chosen.

Table 1. Test functions

app.	name	reference	dim.	opt.
f_1	Sphere model	[11]	var.	0
f_2	Generalized Rosenbrock's function	[11]	var.	0
f_3	Step function	[11]	var.	0
f_6	Schwefel's double sum (function 1.2)	[14, 13]	var.	0
f_9	Ackley's function	[15]	var.	0
f_{15}	Weighted sphere model	[16]	var.	0
f_{16}	Fletcher and Powell const.	[17]	5	0
f_{18}	Shekel-5	[18]	4	-10.1532
f_{19}	Shekel-7	[18]	4	-10.4029
f_{20}	Shekel-10	[18]	4	-10.5364
f_{21}	Griewangk-n2	[18]	var.	0
f_{22}	Griewangk-n10	[18]	var.	0
f_{23}	Galar	[19]	var.	2.00686
f_{24}	Kowalik	[20, 13]	4	0.0003075
q_1	Hyperellipsoid, parallel to axes	[21]	var.	0
q_2	Hyperellipsoid, randomly oriented	[21]	var.	0
sdp	Sum of different powers	[22]	var.	0

5 Test Series

For each function of the test bed, two variants of the NG-EN were compared to the SOM-EN[1]. The parameter settings are given in Table 2. These settings are

Table 2. Parameter settings

strategy	μ	n_O	$t_{max} = \gamma_{max}$	δ	σ_i	σ_f	δ_{min}	δ_{max}	ϵ_i	ϵ_f
SOM-EN	10	4	30000	0.01	-	-	-	-	-	-
NG-EN	10	4	30000	-	5	0.05	1e-5	0.02	0.02	0.01
NG-EN	10	4	30000	-	5	0.05	1e-5	0.02	0.0075	0.00375

based on previous parameter studies made for NG-EN [24] and on experience in using SOM-EN [8].

The results were averaged out over 20 runs.

Table 3 summarizes the results of the test series. For each optimization task (column *app*), the table shows how the three tested strategies (grid-neighborhood, neural-gas neighborhood with $\epsilon_i = 0.02$, $\epsilon_f = 0.01$ and neural-gas neighborhood with $\epsilon_i = 0.0075$, $\epsilon_f = 0.00375$) perform according to the number of function evaluations needed to reach the optimum (*# evals*) and according to how often the method converged, i. e. how often the optimum was found (*# conv.*). The best results are indicated by ($\bullet$).

Due to the preliminary limited number of generations, none of the strategies was able to find the optimum (defined as $f_{\text{opt}}(\boldsymbol{x}) = $ 1e-10) of the applications f_2, q_1, q_2 and *sdp* in time (although for e.g. *sdp* a value $f(\boldsymbol{x}) = $ 1e-8) could be reached). Therefore, no comparisons are shown for those test functions.

Concerning the convergence velocity, it can be seen that the NG-EN with $\epsilon_i = 0.02$ is almost always the fastest, the only exception is function f_3. For most test functions, it is considerably faster than SOM-EN, outperforming SOM-EN by a factor ≈ 2 (f_1, f_{21}, f_{22}, f_{23}), factor ≈ 3 (f_9, f_{16}) or even factor ≈ 5 (f_{19}). The parameter setting $\epsilon_i = 0.0075$, however, results in a noticeably slower convergence. It is almost slower than NG-EN with $\epsilon_i = 0.02$ (again, f_3 is the only exception), and often comparable to or a bit slower than SOM-EN. For the test functions f_9, f_{16}, f_{19} and f_{24}, it is by a factor $\approx 1.5 - 3$ faster than SOM-EN.

Having a look at the robustness of the three strategies, it can be seen that for some test functions all strategies are rather good in finding the optimum. For f_1, f_6 and f_{15} the optimum was always found; for f_{16}, f_{19} and f_{21} all three strategies showed a similar good reliability. But it can also be seen, that in seven out of ten cases, the SOM-EN was more often able to find the optimum. Especially for function f_9 and f_{22} the differing reliability is obvious, while for f_{16} and f_{24} the NG-EN shows better results.

[1] Comparisons of the SOM-EN to conventional ES concerning convergence velocity and robustness can be found in [4] and [23].

Table 3. Comparing NG neighborhood and SOM neighborhood.

app.	**neighb.**	ϵ_i	**# evals**	**# conv.**
f1	grid	-	72010	• 20
f1	n-gas	0.02	• 30010	• 20
f1	n-gas	0.0075	72010	• 20
f3	grid	-	• 503687	• 17
f3	n-gas	0.02	765943	15
f3	n-gas	0.0075	565642	16
f6	grid	-	168010	• 20
f6	n-gas	0.02	• 157210	• 20
f6	n-gas	0.0075	195610	• 20
f9	grid	-	248010	• 15
f9	n-gas	0.02	• 72010	6
f9	n-gas	0.0075	168010	9
f15	grid	-	88810	• 20
f15	n-gas	0.02	• 72816	• 20
f15	n-gas	0.0075	108010	• 20
f16	grid	-	170833	17
f16	n-gas	0.02	• 64210	• 20
f16	n-gas	0.0075	132010	18
f18	grid	-	18676	• 18
f18	n-gas	0.02	• 12010	15
f18	n-gas	0.0075	20010	14
f19	grid	-	54325	• 19
f19	n-gas	0.02	• 11510	16
f19	n-gas	0.0075	18510	16
f20	grid	-	15610	• 20
f20	n-gas	0.02	• 10410	15
f20	n-gas	0.0075	17904	19
f21	grid	-	72010	• 19
f21	n-gas	0.02	• 40010	16
f21	n-gas	0.0075	96010	• 19
f22	grid	-	83086	• 13
f22	n-gas	0.02	• 44454	9
f22	n-gas	0.0075	102010	4
f23	grid	-	23010	• 20
f23	n-gas	0.02	• 12010	13
f23	n-gas	0.0075	29010	18
f24	grid	-	752010	12
f24	n-gas	0.02	• 556010	11
f24	n-gas	0.0075	556245	• 17

Since the functions where NG-EN shows less reliability are multi-modal ones, these observations have led to the assumption, that the NG-EN gets more often stuck in local optima. Due to the predefined grid-neighborhood in SOM-EN, two neighboring individuals in different local optima can still attract each other and pull each other out of the local optimum. In the NG-EN approach, however, the neighborhood is defined depending on the distance of the individuals. Therefore, it is more likely that only those individuals which are stuck in the same local optimum are considered neighbors. Thus, an attraction that leads out of the local optimum is not likely.

6 Conclusions

An evolution strategy with neighborhood attraction (EN) incorporates the neighborhood and learning rule known from neural maps into evolution strategies. Starting from the SOM inspired EN which defines a grid neighborhood between the individuals, an EN with a new neighborhood attraction scheme (NG-EN) was introduced in this paper. The NG-EN dissolves the fixed SOM neighborhood structure and uses a neural gas approach which dynamically adapts the neighborhood structure during learning. The NG-EN approach was compared to the SOM-EN on a basis of exhaustive test series. Although the SOM-EN is still a bit more reliable, especially in solving highly multi-modal optimization tasks with deep and steep local optima, the NG-EN was able to find the optimum of all optimization tasks. It could be shown that for almost all test functions the NG-EN is apparently faster than SOM-EN in finding the optimum.

References

1. Pal, S.K., Mitra, S.: Neuro fuzzy pattern recognition : methods in soft computing. Wiley, New York (1999)
2. Kohonen, T.: Self-Organizing Maps. Springer Verlag, Berlin (1995)
3. Martinetz, T., Schulten, K.: Topology representing networks. Neural Networks **7** (1994)
4. Huhse, J., Zell, A.: Evolution strategy with neighborhood attraction. In Bothe, H., Rojas, R., eds.: Proceedings of the Second ICSC Symposium on Neural Computation – NC 2000, ICSC Academic Press, Canada/Switzerland (2000) 363–369
5. Martinetz, T., Schulten, K.: A "Neural-Gas" network learns topologies. In Kohonen, T., Mäkisara, K., Simula, O., Kangas, J., eds.: Proc. International Conference on Artificial Neural Networks (Espoo, Finland). Volume I., Amsterdam, Netherlands, North-Holland (1991) 397–402
6. Martinetz, T.M., Berkovich, S.G., Schulten, K.J.: 'Neural-gas' network for vector quantization and its application to time-series prediction. IEEE Trans. on Neural Networks **4** (1993) 558–569
7. Villmann, T.: Evolutionary algorithms and neural networks in hybrid systems. In Verleysen, M., ed.: Proceedings of 9th European Symposium on Artificial Neural Networks – ESANN'2001, Evere, Belgium, D-facto (2001)

8. Huhse, J., Zell, A.: Investigating the influence of the neighborhood attraction factor to evolution strategies with neighborhood attraction. In Verleysen, M., ed.: Proceedings of 9th European Symposium on Artificial Neural Networks – ESANN'2001, Evere, Belgium, D-facto (2001) 179–184
9. Schwefel, H.P.: Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie. Birkhäuser, Basel, Stuttgart (1977) Volume 26 of Interdisciplinary Systems Research, German.
10. Altenberg, L.: The evolution of evolvability in genetic programming. In Kinnear, K.E., ed.: Advances in Genetic Programming. Complex Adaptive Systems, Cambridge, MIT Press (1994) 47–74
11. deJong, K.: An analysis of the behaviour of a class of genetic adaptive systems. Master's thesis, University of Michigan (1975)
12. Bäck, T.: A user's guide to genesys 1.0. Technical report, University of Dortmund, Department of Computer Science
13. Schwefel, H.P.: Evolution and Optimum Seeking. John Wiley and Sons, New York (1995)
14. Schwefel, H.P.: Numerical Optimization of Computer Models. Wiley, Chichester (1981)
15. Ackley, D.H.: A connectionist machine for genetic hillclimbing. Kluwer Academic Publishers, Boston (1987)
16. Schwefel, H.P.: Evolutionary learning optimum–seeking on parallel computer architectures. In Sydow, A., Tzafestas, S.G., Vichnevetsky, R., eds.: Proceedings of the International Symposium on Systems Analysis and Simulation 1988, I: Theory and Foundations, Akademie der Wissenschaften der DDR, Akademie-Verlag, Berlin (1988) 217–225
17. Fletcher, R., Powell, M.J.D.: A rapidly convergent descent method for minimization. Comp. J. 6 (1963) 163–168
18. Törn, A., Žilinskas, A.: Global Optimization. Volume 350 of Lecture Notes in Computer Science. Springer, Berlin (1989)
19. Galar, R.: Simulation of local evolutionary dynamics of small populations. Biological Cybernetics **65** (1991) 37–45
20. Schwefel, H.P.: Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie. Volume 26 of Interdisciplinary systems research. Birkhäuser, Basel (1977)
21. Hansen, N., Ostermeier, A., Gawelczyk, A.: Über die Adaptation von allgemeinen, Koordinatensystem-unabhängigen, normalverteilten Mutationen in der Evolutionsstrategie: Die Erzeugendensystemadaption. Technical report, Technische Universität Berlin (1995)
22. Ostermeier, A., Gawelcyk, A., Hansen, N.: A derandomized approach to self-adaptation of evolution strategies. Evolutionary Computation **2** (1995) 369–380
23. Huhse, J., Zell, A.: Evolution strategy with neighborhood attraction - A robust evolution strategy. In Spector, L., Goodman, E.D., Wu, A., Langdon, W.B., Voigt, H.M., Gen, M., Sen, S., Dorigo, M., Pezeshk, S., Garzon, M.H., Burke, E., eds.: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001), San Francisco, California, USA, Morgan Kaufmann (2001) 1026–1033
24. Huhse, J., Villmann, T., Zell, A.: Investigation of the neighborhood attraction evolutionary algorithm based on neural gas. In: Proceedings of the Sixth International Conference on Neural Networks and Soft Computing (ICNNSC) (to appear). Lecture Notes in Computer Science, Springer (2002)