Evolutionary Algorithms in Theory and Practice

Evolution Strategies
Evolutionary Programming
Genetic Algorithms

Dr. Thomas Bäck
Informatik Centrum Dortmund, Germany

New York Oxford
OXFORD UNIVERSITY PRESS
1996

Oxford University Press

Oxford New York
Athens Auckland Bangkok Bombay
Calcutta Cape Town Dar es Salaam Delhi
Florence Hong Kong Istanbul Karachi
Kuala Lumpur Madras Madrid Melbourne
Mexico City Nairobi Paris Singapore
Taipei Tokyo Toronto

and associated companies in
Berlin Ibadan

Published by Oxford University Press, Inc.,
198 Madison Avenue, New York, New York 10016

Library of Congress Cataloging-in-Publication Data
Bäck, Thomas, 1963–
Evolutionary algorithms in theory and practice:
evolution strategies, evolutionary programming,
genetic algorithms/ Thomas Bäck.
p. cm. Includes bibliographical references and index.
ISBN 0-19-509971-0 (hard cover)
1. Genetic algorithms.
2. Evolution (Biology)—Mathematical models.
I. Title.
QA402.5.B333 1995 006.3—dc20 95-13506

1 3 5 7 9 8 6 4 2

Printed in the United States of America
on acid-free paper

In Nature nothing is simple.
Stanislaw Lem (in Likhtenshtein II, p. 132)

Natura non nisi parendo vincitur.
(Nature, to be commanded, must be obeyed.)
Francis Bacon (in Novum Organum, 1620)

3

Artificial Landscapes

In order to facilitate an empirical comparison of the performance of Evolution Strategies, Evolutionary Programming, and Genetic Algorithms, a test environment for these algorithms must be provided in the form of several objective functions $f : \mathbb{R}^n \to \mathbb{R}$. Finding an appropriate and representative set of test problems is not an easy task, since any particular combination of properties represented by a test function does not allow for generalized performance statements. However, there is evidence from a vast number of applications that Evolutionary Algorithms are robust in the sense that they give reasonable performance over a wide range of different topologies.

Here, a set of test functions that are completely artificial and simple is used, i.e., they are stated in a closed, analytical form and have no direct background from any practical application. Instead, they allow for a detailed analysis of certain special characteristics of the topology, e.g. unimodality or multimodality, continuous or discontinuous cases, and others. If any prediction is drawn up for the behavior of Evolutionary Algorithms depending on such strong topological characteristics, the appropriate idealized test function provides a good instrument to test such hypotheses. Furthermore, since many known test sets have some functions in common, at least a minimal level of comparability of results is often guaranteed. Finally, before we can expect an algorithm to be successful in the case of hard problems, it has to demonstrate that it does not fail to work on simple problems.

On the other hand, the (public relations) effect of using artificial topologies is vanishingly small, since the test functions used are of no industrial relevance. This way, researchers working with such test functions can never rest on their industrial laurels.

A more legitimate objection against artificial topologies may be that they are possibly not representative of the "average complexity" of real-world problems, and that some regularity features of their topology may inadmissibly speed up the search. However, most test function sets incorporate even multimodal functions of remarkable complexity, such that only the regularity argument counts against using an artifical function set.

Since we are mainly interested in an objective comparison and verification of some central research hypotheses, a set of artifical test functions seems most appropriate. This is in accordance with many researchers who had the idea to collect sets of test functions with specific properties and to use them as performance benchmarks. In the field of Evolutionary Algorithms, the most famous test sets are due to De Jong (see [DeJ75], pp. 196–210) and Schwefel (see [Sch77], pp. 319–354) for Genetic Algorithms and Evolution Strategies, respectively. Schwefel's test set, consisting of 62 functions that cover an enormous diversity of different topologies (and even highly constrained cases) is surely one of the most extensive function sets. Many other test functions are used in global optimization, and a brief overview of some popular functions is also given in [TŽ89] (pp. 183–186).

De Jong's test function set, consisting of five functions of unimodal and multimodal, continuous and discontinuous, and even noisy character, have been a standard for Genetic Algorithm benchmarks since 1975. Four of these functions are low-dimensional ones, such that especially in case of the only multimodal function (Shekel's foxholes, [She71]) a generalization of results is not possible. An arbitrary scaling of the dimension n is a necessary property of test functions in order to approach reasonable problem sizes (with $n > 20$). Furthermore, as Davis argued, it is quite likely that the location of the optimal solution at the exact middle point of the feasible range in each dimension as chosen by De Jong is advantageous for a Genetic Algorithm due to the high regularity of the bit pattern of the optimum point [Dav91b].

Finally, we mention that artificial test functions in Evolutionary Programming, including e.g. the sphere model, Rosenbrock's function[1] [Ros60], and the multimodal Bohachevsky function [BJS86], were also in most cases only tested with very low dimension [Fog92b].

For our purposes, it is most important to develop a small set of test functions that cover several important features, i.e. the set should

- consist exclusively of functions that are scalable with respect to their dimension n,
- include a unimodal, continuous function for comparison of convergence velocity,

[1] Both the sphere model and Rosenbrock's function are also members of Schwefel's and De Jong's test set. Though being unimodal, the latter function causes difficulties to some optimization methods due to the location of the optimum in a steep parabolic valley with a flat bottom. It caused Rosenbrock to invent his widely used method of rotating coordinates.

- include a step function with several flat plateaus of different height in order to test the behavior of Evolutionary Algorithms in case of absence of any local gradient information,
- cover multimodal functions of different complexity.

The function set does not incorporate pseudoboolean objective functions because Evolution Strategies and Evolutionary Programming are not applicable without major changes of the basic algorithms as introduced in chapter 2. Currently, Genetic Algorithms are useful for a wider range of problems than the two other algorithms described here. This extended generality is achieved by the introduction of an elaborate genotype-phenotype mapping (see section 2.3.1). Recently, Michalewicz presented an empirical investigation of the performance of a *hierarchy* of evolution programs (i.e., a partial ordering of different evolution programs such that the problem ranges to which they can be applied are increasingly general) and indicated that generality grows at the expense of performance [Mic93]. Though it is not intended here to check the validity of this conclusion, our findings in chapter 4 turn out to fit with this "rule".

In the following sections, we will introduce five test functions of our choice in detail, including a three-dimensional[2] graphic of their topology and an explanation of their basic properties. The meaning of "different complexity" as indicated in the last topic of the list presented above will also become clear when multimodal functions are discussed.

3.1 Sphere Model

The sphere model as an example of a continuous, strongly convex, unimodal function was already introduced in connection with the theory of Evolution Strategies in section 2.1.7. It serves as a test case for convergence velocity and is well known and widely used in all fields of Evolutionary Algorithms, occurring in the test sets of Schwefel, De Jong, and Fogel. For algorithms that use the self-adaptation mechanism for mutation step sizes, it provides an appropriate test environment due to the knowledge of optimal step sizes. The three-dimensional (i.e., $n = 2$) topology of the sphere model is shown in figure 3.1.

For reasons of completeness, we present the mathematical description including the default dimension, global minimum point (if known) and its objective function value, and the constraints that define the feasible range (for Genetic Algorithms) for each objective function. Furthermore, notation is redefined now, using f_1 to denote the sphere model:

$$f_1(\vec{x}) = \sum_{i=1}^{n} x_i^2 \tag{3.1}$$

[2]I.e., two object variables ($n = 2$) plus one dimension for the objective function value.

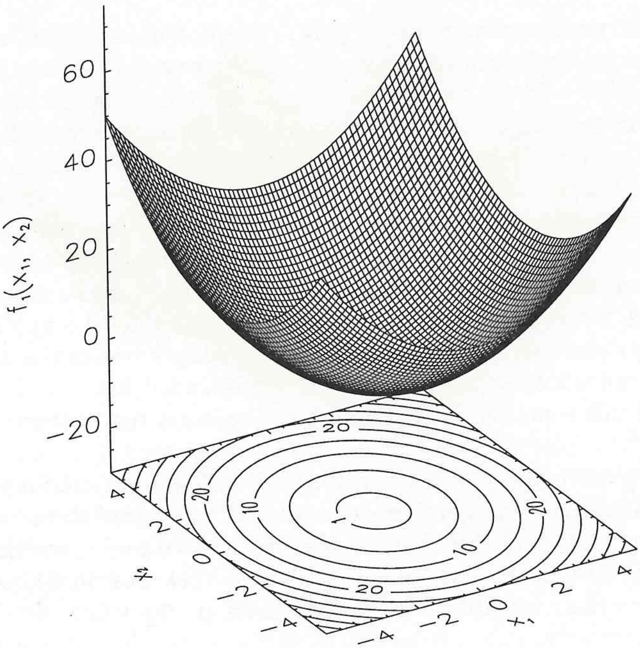

Fig. 3.1: Three-dimensional plot of the sphere model.

$$\vec{x}^* = (0, \ldots, 0)^T \ ; \ f_1^* = 0 \ ; \ n = 30 \ ; \ -40 \leq x_i \leq 60 \ .$$

3.2 Step Function

The background idea of the step function is to make the search more difficult by introducing small plateaus to the topology of an underlying continuous function. While De Jong did this for a simple linear function ([DeJ75], p. 200), we use a discretization of the sphere model obtained by summing over terms[3] $\lfloor x_i + 0.5 \rfloor^2$. As a result, the topology shown in figure 3.2 is discontinuous whenever for any $i \in \{1, \ldots, n\}$ $x_i = k + 0.5$ ($k \in I\!N$).

According to definition 1.5 each plateau corresponds to a local min-

[3] $\lfloor x \rfloor = \max\{i \in \mathbb{Z} \mid i \leq x\}$ denotes the largest integer value less than or equal to x.

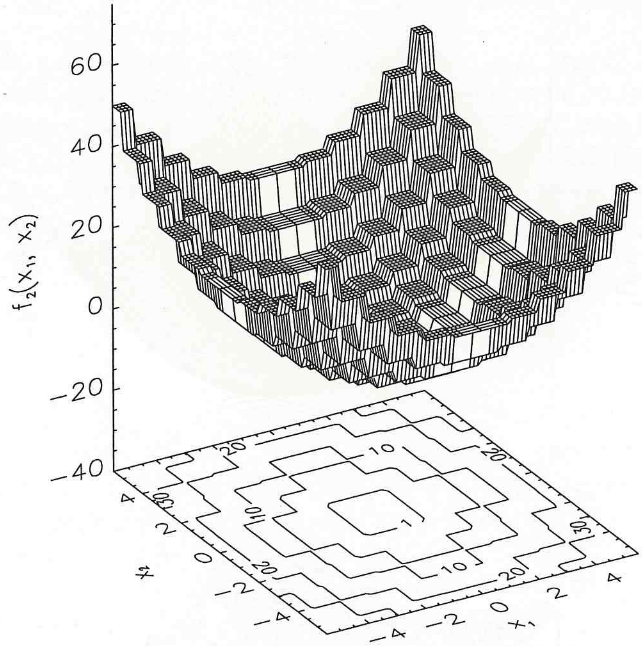

Fig. 3.2: Three-dimensional plot of the step function.

imum of the step function, which therefore is a multimodal function. The plateaus may cause considerable problems to the search process due to the absence of any improvement or worsening of objective function values of offspring located in the close neighborhood of parents, i.e. on a plateau there is no gradient information available for directing the search. Therefore, the search can only be expected to be successful by means of the collective learning process facilitated by a diverse population of solutions.

The global optimum of the step function is located at the plateau having $x_i \in [-0.5, 0.5)$ for all i, and other settings are identical to those for the sphere model, such that altogether the step function is formalized as follows:

$$f_2(\vec{x}) = \sum_{i=1}^{n} \lfloor x_i + 0.5 \rfloor^2 \tag{3.2}$$

$$\vec{x}^* \in ([-0.5, 0.5))^n \; ; \; f_2^* = 0 \; ; \; n = 30 \; ; \; -40 \leq x_i \leq 60 \; .$$

3.3 Ackley's Function

Ackley's function is a continuous, multimodal test function obtained by modulating an exponential function with a cosine wave of moderate amplitude (see [Ack87], pp. 13–14). Originally, it was formulated by Ackley only for the two-dimensional case and is presented here in a generalized, scalable version. Its topology, as shown in figure 3.3, is characterized by an almost flat (due to the dominating exponential) outer region and a central hole or peak where modulations by the cosine wave become more and more influential.

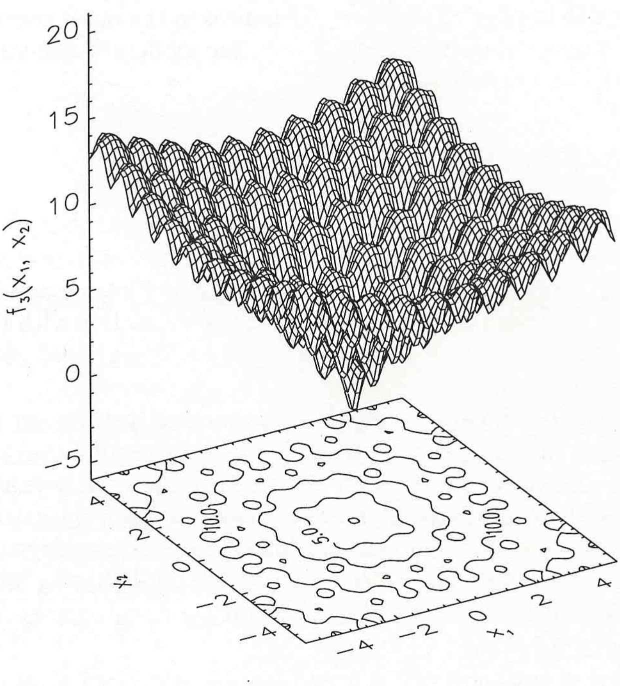

Fig. 3.3: Three-dimensional plot of the generalized function by Ackley.

As Ackley points out, this function causes moderate complications to the search, since though a strictly local optimization algorithm that performs hillclimbing would surely get trapped in a local optimum, a search strategy that scans a slightly bigger neighborhood would be able to cross intervening valleys towards increasingly better optima ([Ack87], p. 14). Therefore, Ackley's function provides a reasonable test case for the necessary combination of path-oriented and volume-oriented characteristics of a search strategy.

To facilitate its use for minimization and to achieve a standardization of the global minimum to an objective function value of zero, the function is formulated as follows:

$$f_3(\vec{x}) = -c_1 \cdot \exp\left(-c_2 \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n} \cdot \sum_{i=1}^{n} \cos(c_3 \cdot x_i)\right) + c_1 + e \tag{3.3}$$

$$c_1 = 20 \; ; \; c_2 = 0.2 \; ; \; c_3 = 2\pi$$

$$\vec{x}^* = (0, \ldots, 0)^T \; ; \; f_3^* = 0 \; ; \; n = 30 \; ; \; -20 \leq x_i \leq 30 \; .$$

In contrast to f_1 and f_2, the feasible range for object variables is smaller due to the fact that fitness differences in the outer regions of the topology become vanishingly small even for moderate dimensions and values for object variables.

3.4 Function after Fletcher and Powell

The highly multimodal function presented here is a typical representative of nonlinear parameter estimation (regression) problems as discussed in section 1.3. It was introduced for the first time by Fletcher and Powell in 1963 (see [FP63]) and also used by Schwefel in connection with Evolution Strategies (see [Sch77], pp. 327–328). A three-dimensional plot of this function is shown in figure 3.4.

In contrast to the other functions discussed in the previous sections, the function f_4 is not symmetric, but instead the extrema are randomly distributed over the search space. This way, the objective function has no implicit symmetry advantage that might simplify optimization for certain algorithms. The random location of extrema is achieved by using random matrices $\mathbf{A} = (a_{ij})$ and $\mathbf{B} = (b_{ij})$ in the following description of the problem:

$$\begin{aligned} f_4(\vec{x}) &= \sum_{i=1}^{n} (A_i - B_i)^2 \\ A_i &= \sum_{j=1}^{n} (a_{ij} \sin \alpha_j + b_{ij} \cos \alpha_j) \\ B_i &= \sum_{j=1}^{n} (a_{ij} \sin x_j + b_{ij} \cos x_j) \end{aligned} \tag{3.4}$$

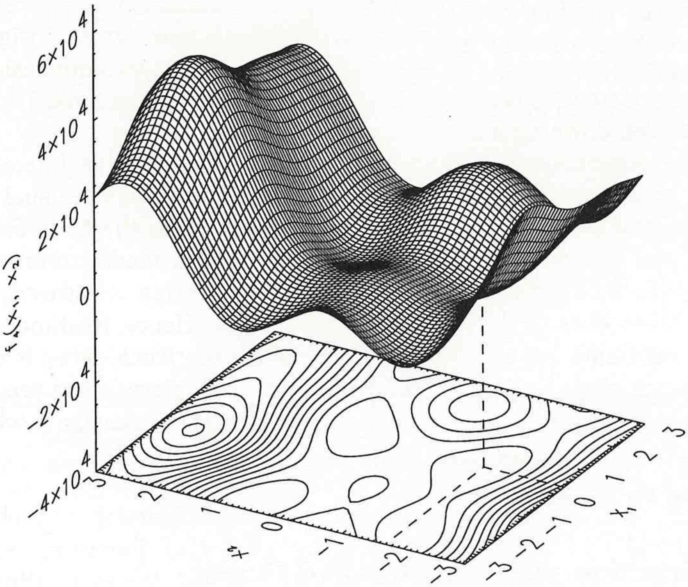

Fig. 3.4: Three-dimensional plot of the function after Fletcher and Powell.

$$\vec{x}^* = \vec{\alpha} \; ; \; f_4^* = 0 \; ; \; n = 30 \; ; \; -\pi \leq x_i \leq \pi$$

$$a_{ij}, b_{ij} \in [-100, 100] \; ; \; \alpha_j \in [-\pi, \pi] \; .$$

As Fletcher and Powell pointed out, there are up to 2^n extrema located in the search interval $|x_i| \leq \pi$. In addition to the matrices **A** and **B**, the vector $\vec{\alpha}$ is also chosen at random. Altogether, these are 1830 random numbers ($n = 30$) which are collected in appendix A.

3.5 Fractal Function

Natural forms that occur e.g. in mountains, canyons, coastlines, and plants are *fractal*, i.e. they show a substantial degree of *self-similarity*. While a mathematical definition requires self-similarity at all resolutions (i.e. the fractal object has no natural, implicit scale) and an infinite, recursive generation process, methods of fractal analysis can also be applied to objects where self-similarity stops at smaller resolution levels. Mountains exhibit their fractal properties by peaks and smaller peaks, rocks, and gravel, coastlines by bays, inlets, estuaries, rivulets, and ditches.

Mandelbrot recognized this fractal structure of nature and developed an underlying mathematical theory [Man83].

For a line segment having dimension one, self-similarity is assured by dividing it into $N \in \mathbb{N}$ equal segments which all look like the original segment scaled down by a factor of $N = N^{1/1}$. For a two-dimensional square, the same is true if it is divided into N parts that look like the original scaled down by a factor of $\sqrt{N} = N^{1/2}$.

Closely associated with the notion of self-similarity is the *fractal dimension* or self-similarity dimension, a generalization of the usual notion of dimensionality. In case of a fractal curve such as the *Koch curve*, however, one can prove that though its total length tends towards infinity if self-similarity on increasingly higher resolution is assured, the resulting curve does not cover all points of a plane. Hence, its dimension is larger than one, but smaller than two. Since the Koch curve is constructed such that dividing it into four pieces each piece is the original curve scaled down by a factor of three, it has fractal dimension d where $4^{1/d} = 3$, i.e. $d = \log(4)/\log(3) \approx 1.26$ (see [Man83], p. 36).

The details of fractal geometry cannot be discussed here, but in order to provide a motivation for introducing a fractal function as well as the basic idea of a fractal and the notion of fractal dimension, some excursion is necessary. Following an idea of Schwefel, the motivation to design a fractal objective function was twofold. First, a fractal function offers the possibility to control its degree of complexity by varying the fractal dimension. As the dimension is increased towards a value of two, the fractal complexity of the topology dominates the surface, while for dimensions close to one the topology is just continuous and smooth. Second, the fractal surface is likely to capture characteristics of noisy real-world objective functions.

The objective function introduced here is obtained by an appropriate modification of the famous *Weierstrass-Mandelbrot function* (e.g. see [Man83], pp. 388–390; [Fed88], pp. 26–30):

$$W(x) = \sum_{j=-\infty}^{\infty} \frac{(1 - \exp(ib^j x)) \cdot \exp(i\varphi_j)}{b^{(2-D)j}} , \qquad (3.5)$$

where i denotes the imaginary unit, $b > 1$ determines how much of the curve is visible for a given range of x, φ_j is an arbitrary phase angle, and D $(1 < D < 2)$ is believed to be the fractal dimension of W (as pointed out in [BL80], this is not yet proved).

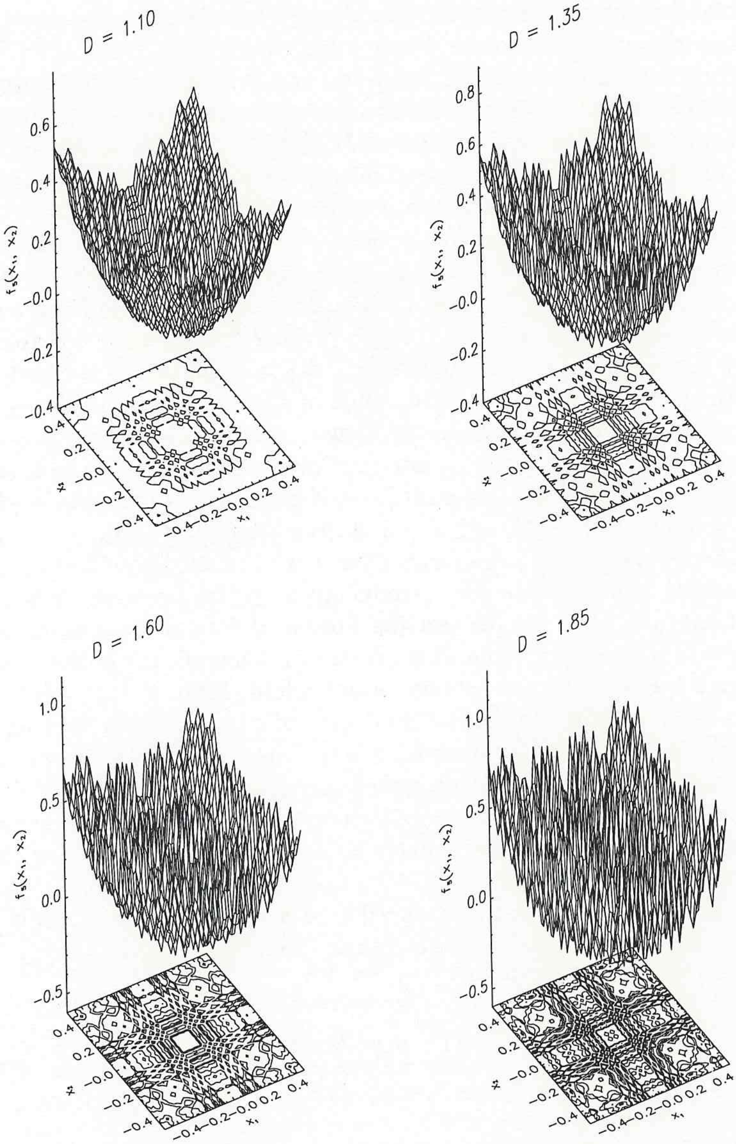

Fig. 3.5: Fractal function for different values of $D \in \{1.1, 1.35, 1.6, 1.85\}$.

From a mathematical point of view, W is a very interesting function because it is continuous but has no derivative at any point. This is also true for the simplified ($\varphi_j = 0$) real part:

$$C(x) = \sum_{j=-\infty}^{\infty} \frac{1-\cos(b^j x)}{b^{(2-D)j}} , \tag{3.6}$$

the Weierstrass-Mandelbrot cosine fractal function[4]. For this function, D is known to be a *box dimension*[5] ([Fed88], p. 27).

As Berry and Lewis indicate, the function $C(x)$ consists of a general underlying trend and a fractal component, but the trend can only be estimated to [BL80]:

$$c(x) \approx \frac{x^{2-D}\Gamma(D-1)\cos(\frac{\pi}{2}(2-D))}{(2-D)\ln b} . \tag{3.7}$$

Since the trend depends on D and therefore hinders comparability of results for different values of D, we removed the trend as exactly as possible, finally superimposing a new trend that is independent of D in order to reintroduce a general shape of the topology. To achieve this, the *self-affinity*[6] of C, $C(bx) = b^{2-D}C(x)$, offers a good opportunity to relate all function values to $C(1)$ and to remove the trend by defining

$$C'(x) = \begin{cases} \dfrac{C(x)}{C(1)|x|^{2-D}} , & \text{if } x \neq 0 \\ 1 , & \text{if } x = 0 . \end{cases} \tag{3.8}$$

Summing up over n such one-dimensional functions, each of which is superimposed by a sphere model trend, we finally obtain

$$f_5(\vec{x}) = \sum_{i=1}^{n} \left(C'(x_i) + x_i^2 - 1\right) \tag{3.9}$$

$$D = 1.85 \; ; \; b = 1.5 \; ; \; n = 20 \; ; \; -5 \leq x_i \leq 5 ,$$

where the constant 1 is subtracted in order to move function values of points close to the origin to an expected value of zero rather than one.

[4]Practically, the infinite sum is approximated as follows to calculate $C(x)$: Starting with $j = 0$ and alternating the sign of j, the summation process continues as long as the relative difference between the last and the actual partial sum exceeds a threshold value $\varepsilon_C = 10^{-8}$ or a maximum number of iterations is reached.

[5]Several different notions of dimension have been introduced to fulfil practical requirements. In case of the box dimension, the minimal number N_s of squares of side length s needed to cover the fractal object completely is counted for smaller and smaller values of s, and finally one obtains

$$D = -\lim_{s\to 0} \frac{\log(N_s)}{\log(s)} .$$

The box dimension D is also known under the term *Minkowski-dimension* (see [Jet89], p. 150).

[6]C is *not* self-similar.

Due to the numerical difficulties of f_5 it is impossible to indicate the exact position of the global minimum.

The three-dimensional topology of the objective function is shown in figure 3.5 for four different values of $D \in \{1.1, 1.35, 1.6, 1.85\}$. The general shape of the function is dominated by the sphere model, and the ruggedness of the surface increases remarkably as the value of the box dimension D is increased. This way, D is a parameter that allows for arbitrarily increasing or decreasing the complexity of this objective function.

3.6 Summary

Five artificial objective functions were introduced in this chapter to serve as a basis for empirical comparisons of Evolutionary Algorithms, i.e. the sphere model (section 3.1) and its step-function version (section 3.2), the generalized function by Ackley (section 3.3), the function after Fletcher and Powell (section 3.4), and a fractal function based on the Weierstrass-Mandelbrot cosine function (section 3.5). The latter three represent highly multimodal topologies that are expected to cause difficulties to the search process with respect to convergence reliability towards the global optimum.

In principle, all functions are arbitrarily scalable concerning the dimension, but in the case of the Fletcher-Powell function we restrict dimension to values up to $n = 30$ (otherwise, larger matrices would have to be presented in appendix A).

Except for the fractal function, all artificial objective functions introduced here are well known and used by several authors in connection with empirical investigations on Evolutionary Algorithms. At least in case of the Fletcher-Powell function and the fractal function we have good reason to claim that they have strong similarities to real-world problems. The next chapter will give a practical impression of both the complexity of the test functions and the problem solving capacity of Evolutionary Algorithms.

Bibliography

[AS65] M. Abramowitz and I. A. Stegun, editors. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* Dover Publications, New York, 1965.

[Ack87] D. H. Ackley. *A connectionist machine for genetic hillclimbing.* Kluwer, Boston, 1987.

[AHU74] A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms.* Addison Wesley, Reading, MA, 1974.

[ASS90] A. N. Antamoshkin, V. N. Saraev, and E. S. Semenkin. Optimization of unimodal monotone pseudoboolean functions. *Kybernetika*, 26(5):432–442, 1990.

[ABN92] B. C. Arnold, N. Balakrishnan, and H. N. Nagaraja. *A First Course in Order Statistics.* Wiley Series in Probability and Mathematical Statistics. Wiley, New York, 1992.

[Atm92] W. Atmar. The philosophical errors that plague both evolutionary theory and simulated evolutionary programming. In Fogel and Atmar [FA92], pages 27–34.

[Bäc92a] Th. Bäck. A User's Guide to *GENEsYs 1.0*. Report of the Systems Analysis Research Group, Universität Dortmund, Fachbereich Informatik, 1992.

[Bäc92b] Th. Bäck. The interaction of mutation rate, selection, and self-adaptation within a genetic algorithm. In Männer and Manderick [MM92], pages 85–94.

[Bäc92c] Th. Bäck. Self–Adaptation in Genetic Algorithms. In F. J. Varela and P. Bourgine, editors, *Proceedings of the First European Conference on Artificial Life*, pages 263–271. The MIT Press, Cambridge, MA, 1992.

[Bäc93] Th. Bäck. Optimal mutation rates in genetic search. In Forrest [For93], pages 2–9.

[BH91] Th. Bäck and F. Hoffmeister. Extended selection mechanisms in genetic algorithms. In Belew and Booker [BB91], pages 92–99.

[BHS91] Th. Bäck, F. Hoffmeister, and H.-P. Schwefel. A survey of evolution strategies. In Belew and Booker [BB91], pages 2–9.

[BHS92] Th. Bäck, F. Hoffmeister, and H.-P. Schwefel. Applications of evolutionary algorithms. Report of the Systems Analysis Research Group SYS–2/92, Universität Dortmund, Fachbereich Informatik, February 1992.

[BRS93] Th. Bäck, G. Rudolph, and H.-P. Schwefel. Evolutionary programming and evolution strategies: Similarities and differences. In D. B. Fogel and W. Atmar, editors, *Proceedings of the 2nd Annual Conference on Evolutionary Programming*, pages 11–22. Evolutionary Programming Society, San Diego, CA, 1993.

[Bak85] J. E. Baker. Adaptive selection methods for genetic algorithms. In Grefenstette [Gre85], pages 101–111.

[Bak87] J. E. Baker. Reducing bias and inefficiency in the selection algorithm. In Grefenstette [Gre87a], pages 14–21.

[Bea93] J. C. Bean. Genetics and random keys for sequences and optimization. Technical Report 92–43, Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI, 1993.

[BN93] J. C. Bean and B. A. Norman. Random keys for job shop scheduling. Technical Report 93–7, Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI, 1993.

[BB91] R. K. Belew and L. B. Booker, editors. *Proceedings of the 4th International Conference on Genetic Algorithms.* Morgan Kaufmann Publishers, San Mateo, CA, 1991.

[BL80] M. V. Berry and Z. V. Lewis. On the Weierstrass-Mandelbrot fractal function. *Proceedings Royal Society of London*, A(370):459–484, 1980.

[Bet81] A. D. Bethke. *Genetic algorithms as function optimizers.* PhD thesis, University of Michigan, 1981. Diss. Abstr. Int. 41(9), 3503B, University Microfilms No. 8106101.

[BEHW89] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the Association for Computing Machinery*, 36(4):929–965, October 1989.

[BJS86] I. O. Bohachevsky, M. E. Johnson, and M. L. Stein. Generalized simulated annealing for function optimization. *Technometrics*, 28(3):209–217, August 1986.

[BGH89] L. B. Booker, D. E. Goldberg, and J. H. Holland. Classifier systems and genetic algorithms. In J. G. Carbonell, editor, *Machine Learning: Paradigms and Methods*, pages 235–282. The MIT Press / Elsevier, 1989.

[Bor78] J. Born. *Evolutionsstrategien zur numerischen Lösung von Adaptationsaufgaben.* Dissertation A, Humboldt-Universität, Berlin, 1978.

[BE91] T. Boseniuk and W. Ebeling. Boltzmann-, Darwin-, and Haeckel-strategies in optimization problems. In Schwefel and Männer [SM91a], pages 430–444.

[Box57] G. E. P. Box. Evolutionary operation: A method for increasing industrial productivity. *Applied Statistics*, VI(2):81–101, 1957.

[BD69] G. E. P. Box and N. P. Draper. *Evolutionary Operation. A Method for Increasing Industrial Productivity.* Wiley, New York, 1969.

[Bre62] H. J. Bremermann. Optimization through evolution and recombination. In M. C. Yovits et al., editor, *Self-Organizing Systems.* Spartan Books, Washington, D.C., 1962.

[BRS65] H. J. Bremermann, M. Rogson, and S. Salaff. Search by evolution. In M. Maxfield, A. Callahan, and L. J. Fogel, editors, *Biophysics and Cybernetic Systems — Proceedings of the 2nd Cybernetic Sciences Symposium*, pages 157–167. Spartan Books, Washington, D.C., 1965.

[BG91] C. L. Bridges and D. E. Goldberg. The nonuniform Walsh-schema transform. In Rawlins [Raw91], pages 13–22.

[BBM92] F. Z. Brill, D. E. Brown, and W. N. Martin. Fast genetic selection of features for neural network classifiers. *IEEE Transactions on Neural Networks*, 3(2):324–328, March 1992.

[Bro58] S. H. Brooks. A discussion of random methods for seeking maxima. *Operations Research*, 6:244–251, 1958.

[CMM84] J. G. Carbonell, R. S. Michalski, and T. M. Mitchell. An overview of machine learning. In R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, editors, *Machine Learning. An Artificial Intelligence Approach*, chapter 1, pages 3–23. Springer, Berlin, 1984.

[CES89] R. A. Caruna, L. J. Eshelman, and J. D. Schaffer. Representation and hidden bias II: Eliminating defining length bias in genetic search via shuffle crossover. In N. S. Sridharan, editor, *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pages 750–755. Morgan Kaufmann Publishers, San Mateo, CA, 1989.

[CS88] R. A. Caruna and J. D. Schaffer. Representation and hidden bias: Gray vs. binary coding for genetic algorithms. In J. Laird, editor, *Proceedings of the 5th International Conference on Machine Learning*, pages 153–161. Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[CHMR87] J. P. Cohoon, S. U. Hedge, W. N. Martin, and D. S. Richards. Punctuated equilibria: A parallel genetic algorithm. In Grefenstette [Gre87a], pages 148–154.

[CMR91a] J. P. Cohoon, W. N. Martin, and D. S. Richards. Genetic algorithms and punctuated equilibria in VLSI. In Schwefel and Männer [SM91a], pages 134–144.

[CMR91b] J. P. Cohoon, W. N. Martin, and D. S. Richards. A multi-population genetic algorithm for solving the K-partition problem on hyper-cubes. In Belew and Booker [BB91], pages 244–248.

[CJ91] R. J. Collins and D. R. Jefferson. Selection in massively parallel genetic algorithms. In Belew and Booker [BB91], pages 249–256.

[Cra85] M. L. Cramer. A representation for the adaptive generation of simple sequential programs. In Grefenstette [Gre85], pages 183–187.

[Dav90] Y. Davidor. Epistasis variance: Suitability of a representation to genetic algorithms. *Complex Systems*, 4:369–383, 1990.

[Dav91a] Y. Davidor. Epistasis variance: A viewpoint on GA-hardness. In Rawlins [Raw91], pages 23–35.

[Dav89] L. Davis. Adapting operator probabilities in genetic algorithms. In Schaffer [Sch89], pages 61–69.

[Dav91b] L. Davis. Bit-climbing, representational bias, and test suite design. In Belew and Booker [BB91], pages 18–23.

[Dav91c] L. Davis, editor. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold, New York, 1991.

[Day90] J. E. Dayhoff. *Neural Network Architectures: An Introduction.* Van Nostrand Reinhold, New York, 1990.

[DG91] K. Deb and D. E. Goldberg. Analyzing deception in trap functions. IlliGAL Report 91009, University of Illinois at Urbana-Champaign, IL, 1991.

[DeJ75] K. A. De Jong. *An analysis of the behaviour of a class of genetic adaptive systems.* PhD thesis, University of Michigan, 1975. Diss. Abstr. Int. 36(10), 5140B, University Microfilms No. 76–9381.

[DeJ85] K. A. De Jong. Genetic algorithms: A 10 year perspective. In Grefenstette [Gre85], pages 169–177.

[DeJ87] K. A. De Jong. On using genetic algorithms to search program spaces. In Grefenstette [Gre87a], pages 210–216.

[DeJ92] K. A. De Jong. Are genetic algorithms function optimizers? In Männer and Manderick [MM92], pages 3–13.

[DeJ93] K. A. De Jong. Genetic algorithms are NOT function optimizers. In Whitley [Whi93], pages 5–17.

[DS93] K. A. De Jong and J. Sarma. Generation gaps revisited. In Whitley [Whi93], pages 19–28.

[Dör77] W. Dörfler. *Mathematik für Informatiker*, volume 1. Carl Hanser, München, 1 edition, 1977.

[Dör78] W. Dörfler. *Mathematik für Informatiker*, volume 2. Carl Hanser, München, 1 edition, 1978.

[EAH91] A. E. Eiben, E. H. L. Aarts, and K. M. Van Hee. Global convergence of genetic algorithms: An infinite Markov chain analysis. In Schwefel and Männer [SM91a], pages 4–12.

[Eig71] M. Eigen. Selforganization of matter and the evolution of biological macromolecules. *Die Naturwissenschaften*, 58(10):465–523, 1971.

[Eig76] M. Eigen. Wie entsteht Information? Prinzipien der Selbstorganisation in der Biologie. *Berichte der Bunsen-Gesellschaft*, 11:1059–1081, 1976.

[EG72] N. Eldredge and S. J. Gould. Punctuated equilibria: An alternative to phyletic gradualism. In T. J. M. Schopf, editor, *Models in Paleobiology*, chapter 5, pages 82–115. Freeman, 1972.

[ECS89] L. J. Eshelman, R. A. Caruna, and J. D. Schaffer. Biases in the crossover landscape. In Schaffer [Sch89], pages 10–19.

[ES93] L. J. Eshelman and J. D. Schaffer. Crossover's niche. In Forrest [For93], pages 9–14.

[Fed88] J. Feder. *Fractals*. Plenum Press, New York, 1988.

[FP63] R. Fletcher and M. J. D. Powell. A rapidly convergent descent method for minimization. *Computer Journal*, 6:163–168, 1963.

[Fog89] T. C. Fogarty. Varying the probability of mutation in the genetic algorithm. In Schaffer [Sch89], pages 104–109.

[Fog88] D. B. Fogel. An evolutionary approach to the traveling salesman problem. *Biological Cybernetics*, 60:139–144, 1988.

[Fog91] D. B. Fogel. *System Identification through Simulated Evolution: A Machine Learning Approach to Modeling*. Ginn Press, Needham Heights, 1991.

[Fog92a] D. B. Fogel. An analysis of evolutionary programming. In Fogel and Atmar [FA92], pages 43–51.

[Fog92b] D. B. Fogel. *Evolving Artificial Intelligence*. PhD thesis, University of California, San Diego, CA, 1992.

[Fog92c] D. B. Fogel. Personal communication, August 1992.

[FA90] D. B. Fogel and W. Atmar. Comparing genetic operators with Gaussian mutations in simulated evolutionary processes using linear systems. *Biological Cybernetics*, 63:111–114, 1990.

[FA92] D. B. Fogel and W. Atmar, editors. *Proceedings of the 1st Annual Conference on Evolutionary Programming.* Evolutionary Programming Society, San Diego, CA, 1992.

[FFA91] D. B. Fogel, L. J. Fogel, and W. Atmar. Meta-evolutionary programming. In R. R. Chen, editor, *Proc. 25th Asilomar Conference on Signals, Systems and Computers*, pages 540–545. Pacific Grove, CA, 1991.

[FFAF92] D. B. Fogel, L. J. Fogel, W. Atmar, and G. B. Fogel. Hierarchic methods in evolutionary programming. In Fogel and Atmar [FA92], pages 175–182.

[Fog62] L. J. Fogel. Autonomous automata. *Industrial Research*, 4:14–19, 1962.

[Fog64] L. J. Fogel. *On the Organization of Intellect.* PhD thesis, University of California, Los Angeles, CA, 1964.

[FOW66] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution.* Wiley, New York, 1966.

[For85] S. Forrest. *Documentation for Prisoner's Dilemma and Norms Programs that use the Genetic Algorithm.* The University of New Mexico, Albuquerque, NM, 1985.

[For93] S. Forrest, editor. *Proceedings of the 5th International Conference on Genetic Algorithms.* Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[Fra92] P. Frankhauser. *Beschreibung der Evolution urbaner Systeme mit der Mastergleichung.* Dissertation, Universität Stuttgart, 1992.

[Fra57a] A. S. Fraser. Simulation of genetic systems by automatic digital computers, I. Introduction. *Australian Journal of Biological Science*, 10:484–491, 1957.

[Fra57b] A. S. Fraser. Simulation of genetic systems by automatic digital computers, II. Effects of linkage on rates of advance under selection. *Australian Journal of Biological Science*, 10:492–499, 1957.

[Fra60] A. S. Fraser. Simulation of genetic systems by automatic digital computers, VI. Epistasis. *Australian Journal of Biological Science*, 13:150–162, 1960.

[Fra62] A. S. Fraser. Simulation of genetic systems. *Journal of Theoretical Biology*, 2:329–346, 1962.

[Fri58] R. M. Friedberg. A learning machine: Part I. *IBM Journal*, 2(1):2–13, January 1958.

[FDN59] R. M. Friedberg, B. Dunham, and J. H. North. A learning machine: Part II. *IBM Journal*, 3(7):282–287, July 1959.

[FD87] C. Fujiko and J. Dickinson. Using the genetic algorithm to generate LISP source code to solve the prisoner's dilemma. In Grefenstette [Gre87a], pages 236–240.

[Fut86] D. J. Futuyma. *Evolutionary Biology.* Sinauer Associates, Sunderland, MA, 1986.

[Fut90] D. J. Futuyma. *Evolutionsbiologie.* Birkhäuser Verlag, Basel, 1990.

[Gal85] R. Galar. Handicapped individua in evolutionary processes. *Biological Cybernetics*, 53:1–9, 1985.

[Gal89] R. Galar. Evolutionary search with soft selection. *Biological Cybernetics*, 60:357–364, 1989.

[Gal91] R. Galar. Simulation of local evolutionary dynamics of small populations. *Biological Cybernetics*, 65:37–45, 1991.

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability — A Guide to the Theory of NP–Completeness.* Freemann & Co., San Francisco, CA, 1979.

[GMSW89] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. *Constrained Nonlinear Programming*, volume 1 of *Handbooks in Operations Research and Management Science*, chapter III. Elsevier, Amsterdam, 1989.

[Gol77] M. J. E. Golay. Sieves for low autocorrelation binary sequences. *IEEE Transactions on Information Theory*, IT–23(1):43–51, 1977.

[Gol82] M. J. E. Golay. The merit factor of long autocorrelation binary sequences. *IEEE Transactions on Information Theory*, IT–28(3):543–549, 1982.

[Gol87] D. E. Goldberg. Simple genetic algorithms and the minimal, deceptive problem. In L. Davis, editor, *Genetic algorithms and simulated annealing*, pages 74–88. Pitman, London, 1987.

[Gol89a] D. E. Goldberg. *Genetic algorithms in search, optimization and machine learning.* Addison Wesley, Reading, MA, 1989.

[Gol89b] D. E. Goldberg. Sizing populations for serial and parallel genetic algorithms. In Schaffer [Sch89], pages 70–79.

[Gol90] D. E. Goldberg. A note on Boltzmann tournament selection for genetic algorithms and population–oriented simulated annealing. *Complex Systems*, 4:445–460, 1990.

[GD91] D. E. Goldberg and K. Deb. A comparative analysis of selection schemes used in genetic algorithms. In Rawlins [Raw91], pages 69–93.

[Goo88] R. Goodman. *Introduction to Stochastic Models.* The Benjamin/Cummings Publishing Company, Menlo Park, CA, 1988.

[GS89] M. Gorges-Schleuter. ASPARAGOS: an asynchronous parallel genetic optimization strategy. In Schaffer [Sch89], pages 422–427.

[GS90] M. Gorges-Schleuter. *Genetic Algorithms and Population Structures — A Massively Parallel Algorithm.* Dissertation, Universität Dortmund, 1990.

[GS92] M. Gorges-Schleuter. Comparison of local mating strategies in massively parallel genetic algorithms. In Männer and Manderick [MM92], pages 553–562.

[Got89] W. Gottschalk. *Allgemeine Genetik.* Georg Thieme Verlag, Stuttgart, 3 edition, 1989.

[Gou82] S. J. Gould. Darwinism and the expansion of evolutionary theory. *Science*, 216:380–387, April 1982.

[GKP89] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics.* Addison Wesley, Reading, MA, 1989.

[Gre85] J. J. Grefenstette, editor. *Proceedings of the 1st International Conference on Genetic Algorithms and Their Applications.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1985.

[Gre86] J. J. Grefenstette. Optimization of control parameters for genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics*, SMC–16(1):122–128, 1986.

[Gre87a] J. J. Grefenstette, editor. *Proceedings of the 2nd International Conference on Genetic Algorithms and Their Applications.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1987.

[Gre87b] J. J. Grefenstette. *A User's Guide to GENESIS.* Navy Center for Applied Research in Artificial Intelligence, Washington, D. C., 1987.

[GB89] J. J. Grefenstette and J. E. Baker. How genetic algorithms work: A critical look at implicit parallelism. In Schaffer [Sch89], pages 20–27.

[dGWH89] C. de Groot, D. Würtz, and K. H. Hoffman. Low autocorrelation binary sequences: Exact enumeration and optimization by evolution strategies. Technical Report 89–09, Eidgenössische Technische Hochschule Zürich, November 1989.

[Hah93] S. Hahn. Optimierung von Monte-Carlo-Generator-Parametern mit Genetischen Algorithmen. Diplomarbeit, Bergische Universität, Gesamthochschule Wuppertal, Fachbereich Physik, 1993.

[HBH92] S. Hahn, K. H. Becks, and A. Hemker. Optimizing monte carlo generator parameters using genetic algorithms. In D. Perret-Gallix, editor, *New Computing Techniques in Physics Research II — Proceedings 2nd International Workshop on Software Engineering, Artificial Intelligence and Expert Systems for High Energy and Nuclear Physics*, pages 255–265. World Scientific, Singapore, 1992.

[HB91] W. E. Hart and R. K. Belew. Optimizing an arbitrary function is hard for the genetic algorithm. In Belew and Booker [BB91], pages 190–195.

[Har90] R. F. Hartl. A global convergence proof for a class of genetic algorithms. Technische Universität Wien, 1990.

[HEK82] J. Hartung, B. Elpelt, and K.-H. Klösener. *Statistik — Lehr- und Handbuch der angewandten Statistik.* R. Oldenbourg Verlag, München, 1982.

[Her92] M. Herdy. Reproductive isolation as strategy parameter in hierarchically organized evolution strategies. In Männer and Manderick [MM92], pages 207–217.

[HM91] J. Hesser and R. Männer. Towards an optimal mutation probability in genetic algorithms. In Schwefel and Männer [SM91a], pages 23–32.

[HM92] J. Hesser and R. Männer. Investigation of the M-heuristic for optimal mutation probabilities. In Männer and Manderick [MM92], pages 115–124.

[Hof91] F. Hoffmeister. Scalable parallelism by evolutionary algorithms. In M. Grauer and D. B. Pressmar, editors, *Parallel Computing and Mathematical Optimization*, volume 367 of *Lecture Notes in Economics and Mathematical Systems*, pages 177–198. Springer, Berlin, 1991.

[HB92] F. Hoffmeister and Th. Bäck. Genetic self–learning. In F. J. Varela and P. Bourgine, editors, *Proceedings of the First European Conference on Artificial Life*, pages 227–235. The MIT Press, Cambridge, MA, 1992.

[HS90a] F. Hoffmeister and H.-P. Schwefel. KORR 2.1 — An Implementation of a $(\mu \overset{+}{,} \lambda)$ – Evolution Strategy. Report of the Systems Analysis Research Group, Universität Dortmund, Fachbereich Informatik, November 1990.

[HS90b] F. Hoffmeister and H.-P. Schwefel. A taxonomy of parallel evolutionary algorithms. In G. Legendi, T. Schendel, and U. Wolf, editors, *ParCella 90*, volume 2 of *Research in Informatics*, pages 97–107. Akademie–Verlag, Berlin, 1990.

[Hol62] J. H. Holland. Outline for a logical theory of adaptive systems. *Journal of the Association for Computing Machinery*, 3:297–314, 1962.

[Hol75] J. H. Holland. *Adaptation in natural and artificial systems.* The University of Michigan Press, Ann Arbor, MI, 1975.

[HJ58] R. Hooke and T. A. Jeeves. Comments on Brooks' discussion of random methods. *Operations Research*, 6:881–882, 1958.

[INM88] INMOS Limited. *The Transputer Databook*, Bath Press, 1988.

[JMRS89] A. Jankowski, Z. Michalewicz, Z. W. Ras, and D. Shoff. Issues on evolution programming. In R. Janicki and W. W. Koczkodaj, editors, *Computing and Information*, pages 459–463. Elsevier, Amsterdam, 1989.

[Jan84] E. Jantsch. *Die Selbstorganisation des Universums — Vom Urknall zum menschlichen Geist.* Deutscher Taschenbuch Verlag, München, 2 edition, 1984.

[Jet89] G. Jetschke. *Mathematik der Selbstorganisation: Qualitative Theorie deterministischer und stochastischer dynamischer Systeme.* Vieweg, Wiesbaden, 1989.

[JH91] T. Johnson and P. Husbands. System identification using genetic algorithms. In Schwefel and Männer [SM91a], pages 85–89.

[Kni92] F. Knickmeier. Auslegung eines Rechnernetzwerkes mit minimalem Kommunikationsaufwand mittels evolutionärer Algorithmen. Diplomarbeit, Universität Dortmund, Fachbereich Informatik, 1992.

[Koz89] J. R. Koza. Hierarchical genetic algorithms operating on populations of computer programs. In N. S. Sridharan, editor, *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pages 768–774. Morgan Kaufmann Publishers, San Mateo, CA, 1989.

[Koz90] J. R. Koza. Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems. Technical Report STAN–CS–90–1314, Department of Computer Science, Stanford University, Stanford, CA, 1990.

[Koz91] J. R. Koza. A hierarchical approach to learning the boolean multiplexer function. In Rawlins [Raw91], pages 171–192.

[Koz92] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* Complex Adaptive Systems. The MIT Press, Cambridge, MA, 1992.

[Kre65] E. Kreyszig. *Statistische Methoden und ihre Anwendungen.* Vandenhoeck & Ruprecht, Göttingen, 1965.

[KFM71] P. Krolak, W. Felts, and G. Marble. A man-machine approach towards solving the traveling salesman problem. *Communications of the ACM*, 14(5):327–334, 1971.

[Küp90] B.-O. Küppers. *Der Ursprung biologischer Information.* Piper, München, 2 edition, 1990.

[Kur91] F. Kursawe. A variant of Evolution Strategies for vector optimization. In Schwefel and Männer [SM91a], pages 193–197.

[Kur92] F. Kursawe. Evolution strategies for vector optimization. In G.-H. Tzeng and P. L. Yu, editors, *Preliminary Proceedings of the Tenth International Conference on Multiple Criteria Decision Making*, pages 187–193. National Chiao Tung University, Taipei, 1992.

[vLA87] P. J. M. van Laarhoven and E. H. L. Aarts. *Simulated Annealing: Theory and Applications.* Mathematics and its Applications. Kluwer, Dordrecht, 1987.

[Lan89] C. G. Langton. Artificial life. In C. G. Langton, editor, *Artificial Life*, volume VI of *Studies in the Science of Complexity*, pages 1–48. Santa Fe Institute, Los Alamos, NM, Addison-Wesley, 1989.

[Lic65] H. J. Lichtfuss. Evolution eines Rohrkrümmers. Diplomarbeit, Technische Universität Berlin, 1965.

[LV90] G. E. Liepins and M. D. Vose. Representational issues in genetic optimization. *Journal of Experimental and Theoretical Artificial Intellgence*, 2:101–115, 1990.

[Liu68] C. L. Liu. *Introduction to Combinatorial Mathematics.* Computer Science Series. McGraw-Hill, New York, 1968.

[Loh91] R. Lohmann. *Bionische Verfahren zur Entwicklung visueller Systeme.* Dissertation, Technische Universität Berlin, 1991.

[Loh92a] R. Lohmann. Structure evolution and incomplete induction. In Männer and Manderick [MM92], pages 175–185.

[Loh92b] R. Lohmann. Structure evolution in neural systems. In B. Soucek and the IRIS Group, editors, *Dynamic, Genetic, and Chaotic Programming*, pages 395–411. Wiley, New York, 1992.

[MM92] R. Männer and B. Manderick, editors. *Parallel Problem Solving from Nature 2.* Elsevier, Amsterdam, 1992.

[Mah91] S. Mahfoud. An analysis of Boltzmann tournament selection. IlliGAL Report 91007, University of Illinois at Urbana-Champaign, IL, 1991.

[Man83] B. B. Mandelbrot. *The fractal geometry of nature.* Freeman, New York, 1983.

[May88] E. Mayr. *Toward a new Philosophy of Biology: Observations of an Evolutionist.* The Belknap Press of Harvard University Press, Cambridge, MA, and London, GB, 1988.

[Mic92] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Artificial Intelligence. Springer, Berlin, 1992.

[Mic93] Z. Michalewicz. A hierarchy of evolution programs: An experimental study. *Evolutionary Computation*, 1(1):51–76, 1993.

[MJ91] Z. Michalewicz and C. Z. Janikow. Handling constraints in genetic algorithms. In Belew and Booker [BB91], pages 151–157.

[Mic86] R. S. Michalski. Understanding the nature of learning: Issues and research directions. In R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, editors, *Machine Learning Vol. II*, chapter 1, pages 3–25. Morgan Kaufmann Publishers, San Mateo, CA, 1986.

[MP69] M. Minsky and S. Papert. *Perceptrons.* The MIT Press, Cambridge, MA, 1969.

[Müh92] H. Mühlenbein. How genetic algorithms really work: I. Mutation and hillclimbing. In Männer and Manderick [MM92], pages 15–25.

[MSV93] H. Mühlenbein and D. Schlierkamp-Voosen. Predictive models for the breeder genetic algorithm. *Evolutionary Computation*, 1(1):25–49, 1993.

[MSB91] H. Mühlenbein, M. Schomisch, and J. Born. The parallel genetic algorithm as function optimizer. In Belew and Booker [BB91], pages 271–278.

[MK87] K. G. Murty and S. N. Kabadi. Some NP-complete problems in quadratic and nonlinear programming. *Mathematical Programming*, 1987.

[NS89] M. Nowak and P. Schuster. Error thresholds of replication in finite populations: Mutation frequencies and the onset of Muller's ratchet. *Journal of theoretical Biology*, 137:375–395, 1989.

[OSH87] I. M. Oliver, D. J. Smith, and J. R. C. Holland. A study of permutation crossover operators on the traveling salesman problem. In Grefenstette [Gre87a], pages 224–230.

[PR92] S. E. Page and D. W. Richardson. Walsh functions, schema variance, and deception. *Complex Systems*, 6:125–135, 1992.

[PS82] Ch. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization*. Prentice Hall, 1982.

[Per91] Perihelion Software. *The Helios Parallel Operating System*, 1991.

[Pet91] E. Peters. *Ein Beitrag zur wissensbasierten Auswahl und Steuerung von Optimierverfahren*. Dissertation, Universität Dortmund, 1991.

[PL89] Ch. C. Pettey and M. R. Leuze. A theoretical investigation of a parallel genetic algorithm. In Schaffer [Sch89], pages 398–405.

[PLG87] Ch. C. Pettey, M. R. Leuze, and J. J. Grefenstette. A parallel genetic algorithm. In Grefenstette [Gre87a], pages 155–161.

[Pin84] J. Pintér. Convergence properties of stochastic optimization procedures. *Math. Operationsforschung und Statist., Ser. optimization*, 15(3):405–427, 1984.

[Rad90] N. J. Radcliffe. *Genetic Neural Networks on MIMD Computers*. PhD thesis, University of Edinburgh, Scotland, 1990.

[Raw91] G. J. E. Rawlins, editcr. *Foundations of Genetic Algorithms*. Morgan Kaufmann Publishers, San Mateo, CA, 1991.

[Ray91] Th. S. Ray. Is it alive or is it GA? In Belew and Booker [BB91], pages 527–534.

[Rec65] I. Rechenberg. Cybernetic solution path of an experimental problem. Royal Aircraft Establishment, Library translation No. 1122, Farnborough, Hants., UK, August 1965.

[Rec73] I. Rechenberg. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann–Holzboog, Stuttgart, 1973.

[Rec94] I. Rechenberg. *Evolutionsstrategie '94*, volume 1 of *Werkstatt Bionik und Evolutionstechnik*. frommann–holzboog, Stuttgart, 1994.

[Rén77] A. Rényi. *Wahrscheinlichkeitsrechnung*. VEB Deutscher Verlag der Wissenschaften, Leipzig, 1977.

[Ric88] E. Rich. *Artificial Intelligence*. Computer Science Series. McGraw-Hill, 8 edition, 1988.

[Ros58] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[Ros60] H. H. Rosenbrock. An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3):175–184, 1960.

[Rud90] G. Rudolph. Globale Optimierung mit parallelen Evolutionsstrategien. Diplomarbeit, Universität Dortmund, Fachbereich Informatik, 1990.

[Rud91] G. Rudolph. Global optimization by means of distributed evolution strategies. In Schwefel and Männer [SM91a], pages 209–213.

[Rud92a] G. Rudolph. On correlated mutations in evolution strategies. In Männer and Manderick [MM92], pages 105–114.

[Rud92b] G. Rudolph. Parallel approaches to stochastic global optimization. In W. Joosen and E. Milgrom, editors, *Parallel Computing: From Theory to Sound Practice, Proceedings of the European Workshop on Parallel Computing*, pages 256–267. IOS Press, Amsterdam, 1992.

[Rud94] G. Rudolph. Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Networks, Special Issue on Evolutionary Programming*, 1994. (In print).

[Sam59] A. L. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal*, 3:211–229, 1959.

[Sam67] A. L. Samuel. Some studies in machine learning using the game of checkers. II — Recent progress. *IBM Journal*, 11(6):601–617, 1967.

[Sch89] J. D. Schaffer, editor. *Proceedings of the 3rd International Conference on Genetic Algorithms and Their Applications.* Morgan Kaufmann Publishers, San Mateo, CA, 1989.

[SCED89] J. D. Schaffer, R. A. Caruna, L. J. Eshelman, and R. Das. A study of control parameters affecting online performance of genetic algorithms for function optimization. In Schaffer [Sch89], pages 51–60.

[SE91] J. D. Schaffer and L. J. Eshelman. On crossover as an evolutionary viable strategy. In Belew and Booker [BB91], pages 61–68.

[SM87] J. D. Schaffer and A. Morishima. An adaptive crossover distribution mechanism for genetic algorithms. In Grefenstette [Gre87a], pages 36–40.

[SB93] M. Schütz and Th. Bäck. A User's Guide to EVOS 1.0. Report of the Systems Analysis Research Group, Universität Dortmund, Fachbereich Informatik, 1993.

[Sch91] J. Schull. The view from the adaptive landscape. In Schwefel and Männer [SM91a], pages 415–427.

[Sch65] H.-P. Schwefel. *Kybernetische Evolution als Strategie der experimentellen Forschung in der Strömungstechnik.* Diplomarbeit, Technische Universität Berlin, 1965.

[Sch68] H.-P. Schwefel. Projekt MHD-Staustrahlrohr: Experimentelle Optimierung einer Zweiphasendüse, Teil I. Technischer Bericht 11.034/68, 35, AEG Forschungsinstitut, Berlin, October 1968.

[Sch75a] H.-P. Schwefel. Binäre Optimierung durch somatische Mutation. Technischer Bericht, Technische Universität Berlin und Medizinische Hochschule Hannover, May 1975.

[Sch75b] H.-P. Schwefel. *Evolutionsstrategie und numerische Optimierung.* Dissertation, Technische Universität Berlin, May 1975.

[Sch77] H.-P. Schwefel. *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie*, volume 26 of *Interdisciplinary Systems Research.* Birkhäuser, Basel, 1977.

[Sch80] H.-P. Schwefel. Subroutines EVOL, GRUP, KORR, listings and user's guide. Interner Bericht KFA-STE-IB-2/80, Kernforschungsanlage Jülich, Programmgruppe Systemforschung und Technologische Entwicklung, April 1980.

[Sch81a] H.-P. Schwefel. *Numerical Optimization of Computer Models.* Wiley, Chichester, 1981.

[Sch81b] H.-P. Schwefel. Optimum seeking methods: Subroutines for the minimization of non–linear functions of several variables by means of direct (derivative–free) methods. Interner Bericht KFA-STE-IB-7/81, Kernforschungsanlage Jülich, October 1981.

[Sch87] H.-P. Schwefel. Collective phenomena in evolutionary systems. In *Preprints of the 31st Annual Meeting of the International Society for General System Research, Budapest*, volume 2, pages 1025–1033, June 1987.

[Sch88] H.-P. Schwefel. Evolutionary learning optimum–seeking on parallel computer architectures. In A. Sydow, S. G. Tzafestas, and R. Vichnevetsky, editors, *Proceedings of the International Symposium on Systems Analysis and Simulation 1988, I: Theory and Foundations*, pages 217–225. Akademie-Verlag, Berlin, September 1988.

[Sch95] H.-P. Schwefel. *Evolution and Optimum Seeking.* Sixth-Generation Computer Technoloy Series. Wiley, New York, 1995.

[SB92] H.-P. Schwefel and Th. Bäck. Künstliche Evolution — eine intelligente Problemlösungsstrategie ? *KI*, 6(2):20–27, June 1992.

[SM91a] H.-P. Schwefel and R. Männer, editors. *Parallel Problem Solving from Nature — Proceedings 1st Workshop PPSN I*, volume 496 of *Lecture Notes in Computer Science.* Springer, Berlin, 1991.

[Sel59] O. G. Selfridge. *Pandemonium: A paradigm for learning*, pages 511–527. H.S.M.O., 1959.

[SZ90] R. Serra and G. Zanarini. *Complex systems and cognitive processes.* Springer, Berlin, 1990.

[She71] J. Shekel. Test functions for multimodal search techniques. In *Fifth Annual Princeton Conference on Information Science and Systems*, 1971.

[SW81] F. J. Solis and R. J.-B. Wets. Minimization by random search techniques. *Mathematics of Operations Research*, 6(1):19–30, 1981.

[Spe93] W. M. Spears. Crossover or mutation ? In Whitley [Whi93], pages 221–237.

[SD91a] W. M. Spears and K. A. De Jong. An analysis of multi-point crossover. In Rawlins [Raw91], pages 301–315.

[SD91b] W. M. Spears and K. A. De Jong. On the virtues of parameterized uniform crossover. In Belew and Booker [BB91], pages 230–236.

[SM90] P. Spiessens and B. Manderick. A genetic algorithm for massively parallel computers. In R. Eckmiller, G. Hartmann, and G. Hauske, editors, *Parallel Processing in Neural Systems and Computers*, pages 31–36. North–Holland, 1990.

[SM91b] P. Spiessens and B. Manderick. A massively parallel genetic algorithm: Implementation and first analysis. In Belew and Booker [BB91], pages 279–286.

[Spr93] J. Sprave. Zelluläre evolutionäre Algorithmen zur Parameteroptimierung. In R. Hofestädt, F. Krückeberg, and T. Lengauer, editors, *Informatik in den Biowissenschaften*, Informatik aktuell, pages 111–120. Springer, Berlin, 1993.

[SWM91] T. Starkweather, D. Whitley, and K. Mathias. Optimization using distributed genetic algorithms. In Schwefel and Männer [SM91a], pages 176–186.

[Sum92] B. H. Sumida. Genetics for genetic algorithms. *ACM SIGBIO Newsletter*, 12(2):44–46, June 1992.

[SHMH90] B. H. Sumida, A. I. Houston, J. M. McNamara, and W. D. Hamilton. Genetic algorithms and evolution. *Journal of Theoretical Biology*, 147:59–84, 1990.

[Sys89] G. Syswerda. Uniform crossover in genetic algorithms. In Schaffer [Sch89], pages 2–9.

[Tan87] R. Tanese. Parallel genetic algorithm for a hypercube. In Grefenstette [Gre87a], pages 177–183.

[Tan89] R. Tanese. Distributed genetic algorithms. In Schaffer [Sch89], pages 434–439.

[TŽ89] A. Törn and A. Žilinskas. *Global Optimization*, volume 350 of *Lecture Notes in Computer Science*. Springer, Berlin, 1989.

[TW91] A. Trew and G. Wilson. *Past, Present, Parallel — A Survey of Available Parallel Computing Systems*. Springer, Berlin, 1991.

[Val84] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, November 1984.

[Vee90] B. Veer. *The CDL Guide*. Perihelion Software Ltd., Somerset, UK, 1 edition, January 1990.

[VL91] M. D. Vose and G. E. Liepins. Punctuated equilibria in genetic algorithms. *Complex Systems*, pages 31–44, 1991.

[Whi89] D. Whitley. The GENITOR algorithm and selection pressure: Why rank–based allocation of reproductive trials is best. In Schaffer [Sch89], pages 116–121.

[Whi93] D. Whitley, editor. *Foundations of Genetic Algorithms 2*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[WS90] D. Whitley and T. Starkweather. GENITOR II: A distributed genetic algorithm. *Journal of Experimental and Theoretical Artificial Intelligence*, 2:189–214, 1990.

[Wid62] B. Widrow. Generalization and information storage in networks of adelaine 'neurons'. In M. C. Yovits, G. T. Jacobi, and G. D. Goldstein, editors, *Self-Organizing Systems*, pages 435–461. Spartan Books, Washington, D.C., 1962.

[WK93] Th. Williams and C. Kelley. GNUPLOT – An interactive Plotting Program. C source code Version 3.4, Pixar Corporation, 1993. Available via anon. ftp from host `ftp.germany.eu.net` as file `gnuplot-3.4.tar.Z` in directory `/pub/packages/gnu`.

[Wri91] A. H. Wright. Genetic algorithms for real parameter optimization. In Rawlins [Raw91], pages 205–218.

[Zhi92] A. A. Zhigljavsky. *Theory of Global Random Search*. Mathematics and Its Applications. Kluwer, Dordrecht, 1992.

[ZŽ91] A. A. Zhigljavsky and A. Žilinskas. *Metody Poiska Globalnowo Ekstremuma*. Nauka, Moscow, 1991. (In Russian).