Convergence Behavior of Evolution Strategies on Ridge Functions

Dissertation
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
der Universität Dortmund
am Fachbereich Informatik

von
Ahmet İrfan Oyman

Dortmund
1999

Tag der mündlichen Prüfung: 15. 3. 1999

Dekan: Prof. Dr. Heinrich Müller

Gutachter:
1. Privatdozent Dr. Hans-Georg Beyer
2. Univ.-Prof. Dr. Ingo Wegener

Acknowledgment

Several people and organizations helped me directly or indirectly in the completion of this work. It does not make sense to distinguish them according to the quantity or quality of their support. Nor the list is limited to the people below who I wanted to mention explicitly. I thank them all because of their patience, keenness, and virtue.

My mother Sıdıka Oyman made lovely and calming comments and gave emotional support. My father Mustafa Kemâl Oyman answered my questions in a concise manner. He was not easy to understand but always available and helpful when needed. I learned many things from them both. My brother Eylem İlker Oyman let me feel that I have a brother, a year younger than me: In the high school, in the university, and in later years: the criticizer of my lifestyle. My brother and I thank to our uncle Halil S. Oyman for giving us our first computer (`Amstrad CPC 6128`) as a present.

I thank my teachers in five primary schools I visited. Also my teachers in the high school I visited (*İstanbul Lisesi*) deserve attention and praise. Dieter Hackenberg (my mathematics teacher) was the one who influenced my life at most among them.

I thank the Department of Computer Engineering of the *Boğaziçi University* for the things I learned there (my vocation, among others). Prof. Dr. H. Levent Akın was the supervisor of my BS and MS theses; however, he did more than that: His motivation, support, and humanism made my MS degree possible, and my life thereupon. I will cite from his e-mail signature — "To acquire knowledge and communicate it to others has been the ambition, pleasure, and business of my life.", William Harvey. Prof. Dr. Cem Ersoy and Prof. Dr. Yağmur Denizhan (the latter from the EE department) were in my MS thesis jury. They continued to motivate me thereafter.

Many people helped me in Dortmund since July 1995. The friends from the high school (Alkan Öztürk and Murat Öztekin, among others), and many others I have known here should be mentioned. All members of the Systems Analysis Research Group (also known as CASA, Center of Applied Systems Analysis) provided a convenient working atmosphere.

I thank Ulrich Hermes (our technician) for taking care of the hardware (especially the Sun SPARCstation 10 "`enki`") and the software of the research group. Hilmar Rauhe and Ralf Garionis were the two people who enriched my life with beautiful conversations, mostly during the lunch time. I will thank Peter Dittrich for his comments and for the discussions.

I will especially thank to the people who helped me directly related to my PhD work. Without their generous helps, I would not be able to finish this work. Prof. Dr. Gisbert Dittrich gave me detailed informations and helps on the PhD process. I thank Dr. Hans-

Michael Voigt because of his corrections on the chapter of algorithms. Prof. Dr. Hajime Kita gave me some important ideas for the introduction part. He noticed some technical consequences of the results as well. Prof. Dr. Kalyanmoy Deb made several corrections on the whole work, gave supporting advises, and suggested several developments. Ms. Dipl. Päd. Sigrid Dany made several suggestions and corrections on the style and form of the defense presentation.

Prof. Dr. Claudio Moraga motivated me for a second journal paper and made several corrections on it before the submission. Prof. Dr. Ingo Wegener accepted to be the second reviewer for the thesis jury. I thank him for his interest to my work.

This work would not be possible without generous helps of the DAAD (German Academic Exchange Service), which were both financial (Contract # A/95/11445) and social: I would especially thank Dr. Helmut Blumbach, Ms. Gerda Nellessen-Assenmacher, Ms. Gülseren Salman, and Ms. Annerose Panske.

Prof. Dr. Hans-Paul Schwefel invited me to Dortmund. He helped me in my settlement in Dortmund, he helped also in making realistic plans and assumptions. He introduced me to several important scientists in the research area of evolutionary algorithms. He was my supervisor until the Fall of 1998, and guided me in this period. He proofread and commented on everything I wrote during that time. I thank him for his ideas, academic helps, suggestions, and motivations during this work.

I am thankful to Privatdozent Dr. Hans-Georg Beyer for several reasons. In the beginning, he was my "elder brother", and later he became my thesis supervisor. I applied his methods in the theoretical analysis. He suggested numerous corrections to my work. He has been silent, calm, and helpful. He consistently supported me, much more than he guided. He corrected the nuances in the interpretations of the results.

Lastly, I want to thank my wife Michaela Oyman-Seigner. She was (and is) my hope in darkness, shareholder in happiness, and the sense of my life. She was (and is) patient, supportive, helpful, and kind. She proofread the work to check the grammatical part. I thank her for my new life and pace since Tanabata Matsuri'97.

Dortmund, 29. 11. 1998 Ahmet İrfan Oyman

Contents

List of Figures

List of Tables

Abstract

Convergence Behavior of Evolution Strategies on Ridge Functions

Ahmet İrfan Oyman
University of Dortmund, 1999
Keywords. evolutionary algorithms, evolution strategy, ridge functions, convergence behavior, progress rate, systems analysis.

This work is dedicated to the analysis of the convergence behavior of evolution strategies (ES). The ridge functions are used in the theoretical and empirical analysis. The analysis is carried out both in the fitness space (the space of objective function values) and in the search space (the space of object variables). The ES algorithms are probabilistic optimization algorithms. Therefore, stochastic measures are used to formalize the convergence. The progress rate φ, the quality gain $\overline{Q}$, and the success probabilities P_{s1} and $P_{s\lambda}$ are measures known from the ES literature. Because of the special structure of ridge functions, some additional convergence measures are defined in this work based on the quantity r. This quantity measures the Euclidean distance of a given point in the search space to the ridge axis.

One obtains several different ridge functions depending on the two real-valued parameters in the definition of ridge functions. The ridge functions are symmetrical around their axis. Their optimum is at infinity, but on this axis. Therefore, the progress is measured along the ridge axis. The minimization of r appears as a subgoal, and the distance r has an important role in the analysis. The maximization of the distance traveled along the ridge axis and the minimization of the distance r become conflicting goals if the ridge axis is *not* aligned with a variable axis. This inseparable case is principally considered in the analysis. For a specific ES algorithm, the aligned case has given additional results.

In this work, the convergence measures defined in the search space (i.e. the search space measures) are primarily used in the analysis, since they have given more reliable information on the convergence to the optimum. Additionally, it is investigated whether the fitness space measures are useful in the estimation of the search space measures. In general, such a relation could not be established. Conversely, the quality gain $\overline{Q}$ (a fitness measure) is obtained by an alternative formula of search space measures.

The most important part of this work consists of the derivations of search space mea-

sures using the induced order statistics of relevant probability distributions. The simulation results are used together with these analytical results to examine some hypotheses in the ES literature. The optimum value of the mutation strength for some ridge functions contradicts to the hypothesis on the "evolution window" and to the universal progress law. As another example, the convergence behavior on ridge functions cannot be explained by a "diffusion along the gradient path".

Additionally, the elitist and non-elitist versions of an exemplary ES algorithm are compared under the same conditions for both: The results illuminated other aspects of the convergence behavior. A new form of the evolutionary progress principle is observed on the progress rate formulae of ridge functions. The genetic repair hypothesis is used together with this principle to explain how recombination can help in obtaining larger progress rate values. Some of these selected results may help in the near future to understandthe working mechanisms of multiplicative self-adaptation operators better.

Chapter 1

Introduction

This work attempts to analyze the convergence behavior of evolution strategies (ES) on ridge functions. The analysis is carried out both theoretically and empirically. *Evolution strategy* is a class of "evolutionary algorithms" (EA), which are probabilistic optimization algorithms that have operators inspired from the evolution in the nature. These algorithms are easy to code in computer programs. Since they do not require additional information other than the fitness function values, first and second order partial derivatives of the fitness function are not needed. Therefore, these algorithms are applied in a wide spectrum of optimization problems. Unfortunately, the research on *how* these algorithms work in general is not established in that scale. This work extends the theoretical analysis of ES algorithms to the ridge functions and provides results on the working principles of ES algorithms. If these principles are understood better, the parameters of these algorithms (such as the mutation strength (σ), the number of parents (μ), and the number of descendants (λ)) can be adjusted better. Moreover, further problem-specific algorithms can be developed more effectively.

The analysis of the *convergence behavior* is carried out using the measures defined in the space of fitness values (objective function values) and in the space of object variables (also called search space). These measures represent expected values of respective random variables, or they are defined using the distributions of these variables. Therefore, they are expected to provide an adequate framework for a theoretical investigation. The analysis concentrates on the search space measures since they give direct information on the convergence to the optimum. The fitness space measures are used to estimate the search space measures.

The *ridge function* family is used as the fitness function in the theoretical and empirical analysis. The ridge functions are unimodal. In their simplest form, the optimum lies at infinity for one variable and at zero for all other variables. Unlike the sphere model, this function tests the EA's capability to converge to a distant optimum in the search space. This fitness function demands an algorithm to take larger and larger values in the neighborhood of (and along) a progress axis. By tuning the two parameters of the ridge functions, one can obtain different fitness landscapes. This function family is conjectured to model the fitness landscapes distant from the optimum, i.e. the regions of the search space from which the optimum is not expected to be reached in a few generations. As a result of the structure of the ridge functions, a long term goal and a short term goal will be defined for the analysis. Briefly, the long term goal is formalized as maximizing the single variable along the ridge axis, and the short term goal as minimizing the remaining $N-1$ variables. These two subgoals are not necessarily separable, since the ridge axis can be rotated in the search space. These subgoals are necessary in understanding the operating principles of the ES algorithms on ridge functions. Naturally, these principles will be reflected in the structure of the conclusions.

In the theoretical analysis, a local model is used for the explanation of the search behavior of ES algorithms. Thereafter, many results are obtained analytically using the induced order statistics. The experiments will serve to verify the applicability of the theoretical results. Moreover, additional interesting observations will be outlined in the experimental chapter. In the analysis, the mutation strength σ is assumed to be constant. Therefore, the effect of the self-adaptation on the convergence behavior is to be considered in the future research. Some comments on self-adaptation can be found in Section 8.2. The results obtained in this work will establish a good starting point for this analysis. The formal definitions of the terms used can be found in the respective chapters while the relevant references are given in the index. Therefore, a list of important symbols is not provided.

The remainder of the thesis is organized as follows. Chapter 2 starts with a detailed description of some ES algorithms. Some possible alterations and further extensions are specified. Furthermore, a brief overview on the algorithms related to the ES is given.

The ridge function family and related functions are introduced in Chapter 3. The measures used in the convergence behavior analysis are defined in Chapter 4. These are the progress rate φ, the quality gain $\overline{Q}$, the self-adaptation response ψ (given for completeness), the success probabilities P_{s1} and $P_{s\lambda}$, and the measure r. In the final remarks, some notations are introduced and the three types of the analysis applied are explained.

Chapter 5 summarizes the state of the theoretical research on ES algorithms. This chapter starts with a brief history of the ES, continues with some hypotheses from the ES literature, supplies the background used in the analysis of this work, and gives some important formulae and results. The method applied in the theoretical analysis is also sketched at the end, which uses the induced order statistics of normally distributed functions.

Chapter 6 consists of the analysis of convergence measures. The results on the fitness space measures (the quality gain $\overline{Q}$ and the success probabilities P_{s1} and $P_{s\lambda}$) are presented first. Thereafter, the progress rate φ is obtained using a local model. These results are followed by the ones obtained using induced order statistics. Finally, the theoretical formulae on the distance r to the ridge axis are determined, which will enlighten several aspects of the analysis.

In Chapter 7, the simulation results are compared with the theoretical ones. Some other quantities are analyzed only empirically, since no theoretical results are available to date. Chapter 8 concludes the research results of this study and suggests a number of extensions to this work.

Chapter 2

Algorithms

Evolution strategies (ES) is the generic name for a special type of evolutionary algorithms that search for the object variables of the optimum of a given fitness function. The operators used in this search process are inspired from the evolution in nature. Formal definitions for the search space, optimum, object variable, fitness function, etc. can be found in the chapter for fitness functions (Chapter 3).

This chapter is organized as follows: The five algorithms given in Section 2.1 through Section 2.5 describe the evolutionary operators in their simplest form. Some effective expansions to or alterations in the algorithms presented are mentioned in Section 2.6. Section 2.7 is devoted to the hierarchical ES, i.e. to the meta level search strategy of ES. In Section 2.8, a short categorization of the related search methods is given briefly, as well as the similarities and differences between those and ES. For the short history of the ES, see the chapter dedicated to the state of the research (Chapter 5).

Table 2.1: The five ES algorithms sorted according to their complexity, and their 18 algorithm lines (L0–L17). The first occurrence of an algorithm line is denoted by "$\star$", its modification by "$\bullet$". A simple notation change is indicated by "$\circ$"; and "$\dagger$" means that the corresponding line has reached its final form in this algorithm with respect to these five algorithms. Mutation (L10) and selection (L14) operators are introduced already in the first algorithm, the recombination operator (L9) in Algorithm 4. The last algorithm introduces the self-adaptation operator (L7 and L8), it will be denoted as $(\mu/\rho, \lambda)$-ES with σ-SA.

	A	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
$(1+1)$	1	$\star$	$\star\dagger$	$\star\dagger$	$\star$	$\star\dagger$						$\star$	$\star$		$\star$	$\star$	$\star\dagger$	$\star\dagger$	$\star\dagger$
$(1,\lambda)$	2	$\bullet$			$\circ$		$\star\dagger$					$\circ$	$\circ\dagger$	$\star\dagger$	$\bullet$	$\bullet$			
(μ,λ)	3	$\bullet$			$\bullet$			$\star$				$\bullet$				$\bullet\dagger$			
$(\mu/\rho,\lambda)$	4	$\bullet$						$\bullet\dagger$			$\star$	$\bullet$							
$(\mu/\rho,\lambda)$ (SA)	5	$\bullet$			$\bullet\dagger$				$\star\dagger$	$\star\dagger$	$\circ\dagger$	$\bullet\dagger$			$\bullet\dagger$				

In Section 2.1 through Section 2.5, the lines of the algorithms are *not* numbered consecutively, indicating that some lines will be inserted for more complicated algorithms. The lines shown will either remain unchanged or be altered, preserving their respective line numbers, in order to ease the comparison of the algorithms. Table 2.1 should give an overview for the introduction and alterations of the eighteen lines in these five algorithms.

A few words should be added here for the symbols used that are related to the fitness functions: The number of variables in the fitness function F, which occurs in the explanation of the algorithms, is denoted by N. As to the Algorithms 1–5, it is assumed without loss of generality (w.l.o.g.) that F returns a scalar fitness value for the given variable setting, i.e. $F: \mathbb{R}^N \to \mathbb{R}$. How other types of fitness functions are treated is mentioned in Section 2.6. This work is not directly concerned with constrained optimization (see Subsection 2.6.3).

2.1 The $(1+1)$-ES

The (1+1)-ES is presented as Algorithm 1. This algorithm is also called the two-membered ES. The zeroth line (L0) gives the name of the algorithm. The symbols used in L0 for the identification of the algorithm are explained at the end of the subsection where they appear first. The beginning of the algorithm is marked by L1. Firstly, the generation counter is

L0	<u>**procedure**</u> $(1+1)$-ES
L1	<u>**begin**</u>
L2	$g := 0$
L3	*initialize*$(\mathrm{P}^{(0)})$ $\left[\mathrm{P}^{(0)} := \left(\mathbf{y}_{\mathrm{P}}^{(0)}, F(\mathbf{y}_{\mathrm{P}}^{(0)})\right)\right]$
L4	<u>**while**</u> <u>**not**</u> *terminate*() <u>**do**</u>
L10	$\mathbf{y}_{\mathrm{C}} := \mathit{mutate}(\mathbf{y}_{\mathrm{P}}^{(g)}, \sigma)$
L11	$F_{\mathrm{C}} := F(\mathbf{y}_{\mathrm{C}})$
L13	$\mathrm{C} := (\mathbf{y}_{\mathrm{C}}, F_{\mathrm{C}})$
L14	$\mathrm{P}^{(g+1)} := \mathit{select}(\mathrm{C}, \mathrm{P}^{(g)})$
L15	$g := g + 1$
L16	<u>**od**</u>
L17	<u>**end**</u>

Algorithm 1: The $(1+1)$-ES.

initialized (L2). The starting point is initialized next in L3. This location is named as the initial parent, denoted as $\mathrm{P}^{(0)}$, which consists of its coordinates in the search space, indicated by the vector $\mathbf{y}_{\mathrm{P}}^{(0)}$, and the corresponding fitness function value $F(\mathbf{y}_{\mathrm{P}}^{(0)})$. $\mathrm{P}^{(0)}$ can be chosen by the user of the algorithm, or set up at random.

The loop of generations starts at L4. L16 marks the end of the loop. The loop is repeated until the termination criterion is fulfilled. In the simplest case, it is executed for a fixed number of times (G, total number of generations), and than the algorithm terminates (L17).

The single descendant, called child C in this algorithm, is created in L10 using the variable setting of the parent $\mathbf{y}_{\mathrm{P}}^{(g)}$. The variable setting of the child, $\mathbf{y}_{\mathrm{C}}$, is generated by adding the N-dimensional pseudo-random vector $\mathbf{z}$ to $\mathbf{y}_{\mathrm{P}}^{(g)}$. In other words, the *mutations* are imitated here by $\mathbf{z}$. Each component z_i of each $\mathbf{z}$ is a new pseudo-random sample of the normal distribution, with zero mean and variance σ^2. The exogenous parameter parameter σ is stated by the user. If we denote a pseudo-random sample of the standard normal distribution by $\mathcal{N}$ (having the mean zero and standard deviation one), the mutation operator can be formalized as follows:

$$\mathbf{y}_{\mathrm{C}} := \mathbf{y}_{\mathrm{P}}^{(g)} + \mathbf{z}, \qquad \mathbf{z} := (z_1, z_2, \ldots, z_N)^{\mathrm{T}} \tag{2.1}$$

$$\forall i \in \{1, \ldots, N\}: \quad z_i := \sigma \cdot \mathcal{N} \tag{2.2}$$

$$p(\mathbf{z}) = \left[\sqrt{2\pi}\sigma\right]^{-N} \exp\left(-\frac{1}{2}\frac{\mathbf{z}^{\mathrm{T}}\mathbf{z}}{\sigma^2}\right) \tag{2.3}$$

Note that the same standard deviation σ is used in this work for all components of $\mathbf{z}$. Such mutations are *isotropic*. In the ES terminology, σ is called the *mutation strength.*

The $(1+1)$-ES algorithm is presented here in the simplest possible form. Originally, the $(1+1)$-ES has an external control mechanism for σ based on the fitness value of the child with respect to its parent. This control method is called "1/5-th *success rule*" (Ein Fünftel Regel) [Rec65]. Using that, σ can be controlled externally during the simulation run. It is shortly described in Subsection 2.6.11.

After the creation of C in L10, $\mathbf{y}_{\mathrm{C}}$ is evaluated in L11 using the fitness function F, yielding F_{C}. In L13, C is defined as comprising $\mathbf{y}_{\mathrm{C}}$ and F_{C}.

The fitness function values of $\mathrm{P}^{(g)}$ and its child C are compared in L14. The better one is assigned as $\mathrm{P}^{(g+1)}$, the parent of the next generation. The child substitutes its parent also for the case of fitness equality, enabling a movement in a flat search space. Lastly, the generation counter g is incremented in L15. The lines L1, L2, L4, L15, L16, and L17 in Algorithm 1 remain unchanged for the algorithms considered here in detail; therefore, they are not explained repeatedly.

In the terminology of the ES, this selection method is named as the *plus* strategy, indicating that the parent $\mathrm{P}^{(g)}$ itself is included in the candidate set for $\mathrm{P}^{(g+1)}$. $\mathrm{P}^{(g)}$ is *not* substituted by C if the child has a worse fitness function value; therefore, the plus strategy uses the *elitist* selection.

The *notation* (1+1)-ES stands for "one parent, plus selection strategy, one descendant", respectively, i.e. it states the number of parents and offspring, as well as the selection method.

In the scope of the $(1+1)$-ES, *mutation* and *selection* operators are introduced. This algorithm is the simplest ES, and the one with least memory requirements. Moreover, it is the simplest evolutionary algorithm possible (see Section 2.8).

2.2 The $(1, \lambda)$-ES

The $(1, \lambda)$-ES is shown in Algorithm 2. In this algorithm, the first expansion beyond Algorithm 1 is carried out: Not just one, but λ descendants are generated in a single generation. Furthermore, the other selection method (the *comma* strategy) is introduced. First of all, notations differing from those used in Algorithm 1 are introduced for the parent

L0 **procedure** $(1, \lambda)$-ES
L1 **begin**
L2 $g := 0$
L3   *initialize*$(\mathrm{P}^{(0)})$ $\left[\mathrm{P}^{(0)} := \left(\mathbf{y}^{(0)}, F(\mathbf{y}^{(0)})\right)\right]$
L4   **while not** *terminate*() **do**
L5   **for** $l := 1$ **to** λ **do**
L10 $\tilde{\mathbf{y}}_l := \mathit{mutate}(\mathbf{y}^{(g)}, \sigma)$
L11 $\tilde{F}_l := F(\tilde{\mathbf{y}}_l)$
L12   **od**
L13 $\tilde{\mathbf{P}}^{(g)} := \left\{\forall l \in \{1, \ldots, \lambda\} : \ (\tilde{\mathbf{y}}_l, \tilde{F}_l)\right\}$
L14 $\mathrm{P}^{(g+1)} := \mathit{select}(\tilde{\mathbf{P}}^{(g)})$
L15 $g := g + 1$
L16   **od**
L17 **end**

Algorithm 2: The $(1, \lambda)$-ES.

and the descendants. That is, the variable setting of the parent individual is denoted as $\mathbf{y}^{(g)}$ (for $g = 0$ as $\mathbf{y}^{(0)}$), and the l-th descendant generated as $\tilde{\mathbf{y}}_l$, respectively. Accordingly, notations of fitness function values are changed to $F(\mathbf{y}^{(g)})$ and $F(\tilde{\mathbf{y}}_l)$ (or alternatively $\tilde{F}_l$). These cosmetic changes are made in L3, L10, and L11; further changes in L13 and L14 are explained below.

As already said, λ descendants are generated using the mutation operator in this algorithm. The construct shown in L5 and L12 is introduced for this purpose, indicating that the lines in-between should be executed λ times, for the consecutive integer values between 1 and λ for l. The resulting λ pairs of $(\tilde{\mathbf{y}}_l, \tilde{F}_l)$ constitute the offspring population $\tilde{\mathbf{P}}^{(g)}$ (L13).

Instead of the plus selection strategy introduced in Algorithm 1, the *comma* selection strategy is used here (L14). In this selection scheme, the parent individual gets lost for the next generation; therefore, the individual having the best fitness value in $\tilde{\mathbf{P}}^{(g)}$ is selected as $\mathrm{P}^{(g+1)}$, no matter whether it is worse than the parent $\mathrm{P}^{(g)}$.

The *notation* in L0 stands for "one parent, comma selection strategy, λ descendants". Therefore, as compared to Algorithm 1, another selection strategy and the notion of offspring *population* is introduced here. The notation $(1 + \lambda)$-ES stands for the counterpart with elitist selection, differing only in the selection operator, *select*$(\tilde{\mathbf{P}}^{(g)}, \mathrm{P}^{(g)})$. Therefore, Algorithm 1 is a special case of Algorithm 2, with $\lambda = 1$ and elitist selection. Additionally, the notation "$\overset{+}{,}$", specifically $(1 \overset{+}{,} \lambda)$-ES, is used to mention *both* selection strategies.

The symbol tilde "~" is used to mark the offspring population and the components of the descendants.

The lines L5, L11, and L12 of Algorithm 2 are used unchanged in Algorithms 3–5, whereas L13 is upgraded only for Algorithm 5.

2.3 The (μ, λ)-ES and the $(\mu + \lambda)$-ES

The (μ, λ)-ES is given in Algorithm 3. Differing from Algorithm 2, each generation will consist of μ individuals here, denoted as $\mathbf{P}^{(g)}$. The population $\mathbf{P}^{(g)}$ is defined formally in L3 for $g = 0$, the subscript m is used to identify the μ parent individuals.

The use of $\mathbf{P}^{(g)}$ instead of $\mathrm{P}^{(g)}$ requires several upgrades in Algorithm 2. Firstly, since $\mathbf{P}^{(g)}$ consists of μ individuals, the initialization in L3 should generate μ individuals. The

L0 **procedure** (μ, λ)-ES
L1 **begin**
L2 $\quad g := 0$
L3 $\quad$ *initialize*$(\mathbf{P}^{(0)})$ $\quad \left[\mathbf{P}^{(0)} := \left\{\forall m \in (1, \dots, \mu) : \left(\mathbf{y}_m^{(0)}, F(\mathbf{y}_m^{(0)})\right)\right\}\right]$
L4 $\quad$ **while not** *terminate*() **do**
L5 $\quad\quad$ **for** $l := 1$ **to** λ **do**
L6 $\quad\quad\quad \mathrm{E}_l := \mathit{mate}(\mathbf{P}^{(g)})$
L10 $\quad\quad\quad \tilde{\mathbf{y}}_l := \mathit{mutate}(\mathrm{E}_l, \sigma)$
L11 $\quad\quad\quad \tilde{F}_l := F(\tilde{\mathbf{y}}_l)$
L12 $\quad\quad$ **od**
L13 $\quad\quad \tilde{\mathbf{P}}^{(g)} := \left\{\forall l \in \{1, \dots, \lambda\} : (\tilde{\mathbf{y}}_l, \tilde{F}_l)\right\}$
L14 $\quad\quad \mathbf{P}^{(g+1)} := \mathit{select}(\tilde{\mathbf{P}}^{(g)})$
L15 $\quad\quad g := g + 1$
L16 $\quad$ **od**
L17 **end**

Algorithm 3: The (μ, λ)-ES.

second change is in the way mutations are applied. The descendants are generated around the parents as in Algorithm 2. However, since $\mu > 1$, the individual E_l in $\mathbf{P}^{(g)}$ should be determined first, which will act as the parent in L10. E_l is selected in L6. A special case of the *mating selection* operator is introduced here. In this case, *mate* does not favor any of the μ parents, and selects one of them with probability $1/\mu$:

$$
\begin{aligned}
r &:= \texttt{Uniform}\{1, \dots, \mu\} \ , \qquad \text{therefore} && (2.4)\\
\mathrm{E}_l &:= \left(\mathbf{y}_r^{(g)}, F(\mathbf{y}_r^{(g)})\right) \ , \qquad \text{as a result} && (2.5)\\
\tilde{\mathbf{y}}_l &:= \mathbf{y}_r^{(g)} + \mathbf{z} \ . && (2.6)
\end{aligned}
$$

L6 is implemented by (2.4) and (2.5). First, one of the μ parents is selected randomly according to the discrete uniform distribution. And then, $\tilde{\mathbf{y}}_l$ is generated using its object variables $\mathbf{y}_r^{(g)}$. Equation (2.6) is analogous to (2.1) and corresponds to L10.

The last change is at L14, where the best μ individuals in $\tilde{\mathbf{P}}^{(g)}$ are selected to compose $\mathbf{P}^{(g+1)}$. If we are trying to maximize F, the fitness function values of the λ descendants can be sorted as

$$\tilde{F}_{1:\lambda} \leq \tilde{F}_{2:\lambda} \leq \ldots \leq \tilde{F}_{\lambda-1:\lambda} \leq \tilde{F}_{\lambda:\lambda} \ , \tag{2.7}$$

where $\tilde{F}_{1:\lambda}$ denotes the smallest fitness function value, and $\tilde{F}_{\lambda:\lambda}$ the largest one, respectively. According to the *comma* selection strategy, the individuals with $\tilde{F}_{\iota:\lambda}$ for all $\iota \in \{\lambda - \mu + 1, \ldots, \lambda\}$ will compose $\mathbf{P}^{(g+1)}$. Therefore, the condition $\lambda > \mu$ must be fulfilled to ensure enough offspring.

If the *plus* strategy is concerned, L14 becomes

$$\text{L14} \qquad \mathbf{P}^{(g+1)} := select(\tilde{\mathbf{P}}^{(g)}, \mathbf{P}^{(g)}) \ . \tag{2.8}$$

In this case, the candidate pool has $\gamma := \lambda + \mu$ individuals. The γ fitness function values are sorted as

$$F_{1:\gamma} \leq F_{2:\gamma} \leq \ldots \leq F_{\gamma-1:\gamma} \leq F_{\gamma:\gamma} \ . \tag{2.9}$$

The individuals with $F_{\iota:\gamma}$ for all $\iota \in \{\lambda + 1, \ldots, \lambda + \mu\}$ are selected to compose $\mathbf{P}^{(g+1)}$. Therefore, $\lambda \geq 1$ is sufficient in this case.

The selection operator in L14 is not altered in further algorithms, and is called "*truncation selection*" in the ES terminology. It imitates the livestock breeding. Other selection methods are shortly mentioned in Subsection 2.6.4. In L6, a special case of sexual selection is introduced. The operator *mate* will be expanded further in the next algorithm to select more than one parent per descendant.

Algorithm 3 has more than one parent per generation; therefore, it implements a parental *population*. In L0, (μ, λ) indicates "μ parents, comma selection strategy, λ descendants". Actually, the evaluation of the $\mathbf{y}_m^{(0)}$ settings in L3 yielding the fitness value $F(\mathbf{y}_m^{(0)})$ is not necessary for the comma strategy, since this strategy does not use $F(\mathbf{y}_m^{(0)})$. The $(\mu + \lambda)$-ES would use a different L14, as explained above. In order to obtain the *plus* versions of Algorithm 4 and Algorithm 5, one should just substitute L14 with (2.8). Algorithm 2 is a special case of Algorithm 3 for $\mu = 1$.

2.4 The $(\mu/\rho, \lambda)$-ES

The $(\mu/\rho, \lambda)$-ES is presented as Algorithm 4. The *recombination* operator is introduced as the only difference to Algorithm 3. In this case, ρ parent individuals take part in the generation of one descendant, $\rho \leq \mu$. Firstly, these ρ parents should be determined. This step is similar to L6 in Algorithm 3, which generated a *single* parental individual E_l. Here, the sub-procedure *mate* determines an intermediate population of size ρ. It selects ρ

L0 **procedure** $(\mu/\rho, \lambda)$-ES
L1 **begin**
L2 $\quad g := 0$
L3 $\quad$ *initialize*$(\mathbf{P}^{(0)})$ $\quad \left[\mathbf{P}^{(0)} := \left\{\forall m \in (1, \dots, \mu) : \left(\mathbf{y}_m^{(0)}, F(\mathbf{y}_m^{(0)})\right)\right\}\right]$
L4 $\quad$ **while not** *terminate*() **do**
L5 $\quad\quad$ **for** $l := 1$ **to** λ **do**
L6 $\quad\quad\quad \mathbf{E}_l := \mathit{mate}(\mathbf{P}^{(g)}, \rho)$
L9 $\quad\quad\quad \mathbf{y}_l := \mathit{recombine}(\mathbf{E}_l, \rho)$
L10 $\quad\quad\quad \tilde{\mathbf{y}}_l := \mathit{mutate}(\mathbf{y}_l, \sigma)$
L11 $\quad\quad\quad \tilde{F}_l := F(\tilde{\mathbf{y}}_l)$
L12 $\quad\quad$ **od**
L13 $\quad\quad \tilde{\mathbf{P}}^{(g)} := \left\{\forall l \in \{1, \dots, \lambda\} : (\tilde{\mathbf{y}}_l, \tilde{F}_l)\right\}$
L14 $\quad\quad \mathbf{P}^{(g+1)} := \mathit{select}(\tilde{\mathbf{P}}^{(g)})$
L15 $\quad\quad g := g + 1$
L16 $\quad$ **od**
L17 **end**

Algorithm 4: The $(\mu/\rho, \lambda)$-ES.

parents, that make up together the parent pool or the parental set $\mathbf{E}_l$ for the generation of the l-th descendant. As in (2.4), all parents in $\mathbf{P}^{(g)}$ have the same probability to be selected for $\mathbf{E}_l$. If $\rho = \mu$, $\mathbf{E}_l$ is *per definition* identical to $\mathbf{P}^{(g)}$, and if $\rho < \mu$, any parental individual *may* principally occur more than once in $\mathbf{E}_l$.

The individuals in $\mathbf{E}_l$ are recombined in L9. This step is explained in detail in the following two subsections. The resulting temporary state in the search space is denoted by $\mathbf{y}_l$. The l-th descendant is created around $\mathbf{y}_l$ by the mutation operator in L10, similar to (2.1) and (2.6):

$$\tilde{\mathbf{y}}_l \quad := \quad \mathbf{y}_l + \mathbf{z} \tag{2.10}$$

Algorithm 4 uses more than one parent in the generation of a single descendant. This is expressed with the *notation* in L0, with the meaning "μ parents, ρ parents per individual generated, comma selection strategy, λ descendants". The slash "/" symbol stands for recombination. The type of recombination will also be expressed in this notation, as explained in the following. The special case $\rho = 1$ is the (μ, λ)-ES in Algorithm 3, the case $\rho = 2$ is called *bisexual*, the one $\rho > 2$ *multi-sexual*, and $\rho = \mu$ *panmictic.*

In Algorithm 4, L9 (*recombination*) is introduced, and the lines L6 and L10 are altered to adapt Algorithm 3 for this upgrade. L6 is not altered further in Algorithm 5. The two recombination types analyzed in this work are introduced next.

2.4.1 The $(\mu/\rho_{\mathrm{I}}, \lambda)$-ES

The first one of the two recombination operators analyzed in this work, the *intermediate* recombination, will be introduced here. Actually, the adjective "intermediary" describes the type of the operation better [SR95]; however, it is used less frequently. The special case $\rho = \mu$ will be explained first.

For $\rho = \mu$, we have $\mathbf{E}_l \equiv \mathbf{P}^{(g)}$, as already mentioned. Therefore, all $\mathbf{E}_l$ parent pools for different l are identical; consequently, L6 becomes redundant. Moreover, L9 becomes

$$\text{L9} \qquad \mathbf{y}_l := \textit{recombine}(\mathbf{P}^{(g)}, \mu) \ . \tag{2.11}$$

In this case, L9 calculates the intermediate point of $\mathbf{P}^{(g)}$ in the search space; more formally, their arithmetic mean, center of gravity, or in short, their *centroid*. Consequently, all $\mathbf{y}_l$ are identical for the $\rho = \mu$ case, and the centroid of μ parents can be denoted as $\langle \mathbf{y} \rangle$, and if necessary as $\langle \mathbf{y} \rangle^{(g)}$. Therefore, L9 can be formalized in this case as

$$\langle \mathbf{y} \rangle := \frac{1}{\mu} \sum_{m=1}^{\mu} \mathbf{y}_m^{(g)} \ . \tag{2.12}$$

Since $\langle \mathbf{y} \rangle$ is independent of l, L9 can be executed after L4 and before L5, just *once* per generation g, yielding a structure similar to Algorithm 2.

For the general case $\rho < \mu$, $\mathbf{y}_l$ is the centroid of the $\mathbf{E}_l$ concerned, i.e. of the ρ individuals selected randomly from $\mathbf{P}^{(g)}$. Of course, any parent individual may occur more than once in $\mathbf{E}_l$ for this case.

The *notation* $(\mu/\rho_{\mathrm{I}}, \lambda)$ differs from $(\mu/\rho, \lambda)$ by the subscript $_{\mathrm{I}}$, which denotes that the recombination operator used is the unweighted intermediate one.

2.4.2 The $(\mu/\rho_{\mathrm{D}}, \lambda)$-ES

The second one of the recombination operators analyzed in this work is called the *dominant* recombination. This recombination type is also called "global discrete" in the literature. The object variables of a single individual in the parental set are used predominantly over the others; and this predominant (or prevailing, prominent) individual is sampled anew in the parental set for each variable of $\mathbf{y}_l$. This recombination type is explained in detail next.

In this recombination method, the temporary state $\mathbf{y}_l$ is generated actually in N steps, N being the number of variables. For each variable i, the individual numbered with r is picked randomly from the ρ parents in $\mathbf{E}_l$ (Equation (2.13)). The i-th component of $\mathbf{y}_r^{(g)}$ is used as the i-th component of $\mathbf{y}_l$, i.e. the parent r *dominates* all other parents in $\mathbf{E}_l$ for the component i. Note that r is sampled anew for each variable i, descendant l, and obviously, for each generation g.

For a formal definition of the dominant recombination, $\mathbf{e}_i$ is introduced to denote the unit vector in the direction of the i-th variable in the N dimensional search space. Therefore, the set $\{\forall i \in \{1, \ldots, N\} : \mathbf{e}_i\}$ forms an orthonormal basis of the search space. Consequently, the dominant recombination (L9) is composed of the two steps

$$r_i := \texttt{Uniform}\{1, \ldots, \rho\} \qquad \text{and} \tag{2.13}$$

$$\mathbf{y}_l := \sum_{i=1}^{N} \left(\mathbf{e}_i^{\mathrm{T}} \mathbf{y}_{r_i}^{(g)} \right) \mathbf{e}_i \ . \tag{2.14}$$

Equation (2.13) is similar to (2.4). The subscript $_{\mathrm{D}}$ in $(\mu/\rho_{\mathrm{D}}, \lambda)$ denotes that the recombination operator used is the dominant one.

2.5 The $(\mu/\rho, \lambda)$-ES, with self-adaptation

The $(\mu/\rho, \lambda)$-ES algorithm with self-adaptation of the mutation strength σ (in short: σ-SA) is shown in Algorithm 5. It introduces the lines L7 and L8, dedicated to the endogenous change of the parental mutation strength for the descendant. The lines L3, L9, L10, and L13 of Algorithm 4 are upgraded in order to facilitate this σ-SA mechanism. In the four

L0	**procedure** $(\mu/\rho, \lambda)$-ES, with self-adaptation
L1	**begin**
L2	$\quad g := 0$
L3	$\quad$ *initialize*$(\mathbf{P}^{(0)})$ $\quad \left[\mathbf{P}^{(0)} := \left\{\forall m \in (1, \dots, \mu) : \left(\mathbf{y}_m^{(0)}, s_m^{(0)}, F(\mathbf{y}_m^{(0)})\right)\right\}\right]$
L4	$\quad$ **while** **not** *terminate*() **do**
L5	$\qquad$ **for** $l := 1$ **to** λ **do**
L6	$\qquad\quad \mathbf{E}_l := \mathit{mate}(\mathbf{P}^{(g)}, \rho)$
L7	$\qquad\quad s_l := \mathit{recombine.s}(\mathbf{E}_l, \rho)$
L8	$\qquad\quad \tilde{s}_l := \mathit{mutate.s}(s_l)$
L9	$\qquad\quad \mathbf{y}_l := \mathit{recombine.y}(\mathbf{E}_l, \rho)$
L10	$\qquad\quad \tilde{\mathbf{y}}_l := \mathit{mutate.y}(\mathbf{y}_l, \tilde{s}_l)$
L11	$\qquad\quad \tilde{F}_l := F(\tilde{\mathbf{y}}_l)$
L12	$\qquad$ **od**
L13	$\qquad \tilde{\mathbf{P}}^{(g)} := \left\{\forall l \in \{1, \dots, \lambda\} : (\tilde{\mathbf{y}}_l, \tilde{s}_l, \tilde{F}_l)\right\}$
L14	$\qquad \mathbf{P}^{(g+1)} := \mathit{select}(\tilde{\mathbf{P}}^{(g)})$
L15	$\qquad g := g + 1$
L16	$\quad$ **od**
L17	**end**

Algorithm 5: The $(\mu/\rho, \lambda)$-ES, with self-adaptation.

algorithms introduced up to now, σ is an external (exogenous) parameter, or a strategy parameter, of the ES algorithm. If it is allowed to vary, it becomes a *strategy variable.* In order to express this change, the mutation strength is denoted here as "s" instead of σ.

The σ-SA mechanism will be introduced here, starting with the necessary upgrades beyond Algorithm 4 first. The following subsection (Subsection 2.5.1) is devoted to the actual implementation of the self-adaptation rules, i.e. to some established methods of adapting the mutation strength endogenously (L8).

As already mentioned, the mutation strength is *not* an externally given constant of Algorithm 5, but an endogenous *variable* of this algorithm. Therefore, each individual of $\mathbf{P}^{(g)}$ and $\tilde{\mathbf{P}}^{(g)}$ may have a different mutation strength. Consequently, the definition of an individual must be upgraded accordingly: In Algorithm 5, an individual consists of its variable setting, its fitness value, and the mutation strength used for its generation. Formally, $\mathbf{P}^{(g)}$ is defined in L3 for $g = 0$, and $\tilde{\mathbf{P}}^{(g)}$ in L13.

Beyond the re-definition of $\mathbf{P}^{(g)}$ and $\tilde{\mathbf{P}}^{(g)}$, further changes are necessary in L3, L9, and L10 of Algorithm 4. In L3, μ different mutation strength values, denoted as $s_m^{(0)}$, can be specified for the initial population. The notation for the recombination operator in L9 is changed to *recombine.y*, to contrast the recombination in the domain of the strategy variables (see L7), explained below. L9 is executed in the same way as in Algorithm 4, and operates in the domain of the object variables.

The last change is in L10, where the mutation strength used is explicitly stated in the mutation operator. The notation for the mutation operator itself is also renamed to express that it works in the domain of object variables. It operates as in (2.10), provided that $\tilde{s}_l$ is used as the mutation strength.

Two new lines are introduced in Algorithm 5. The former one, L7, generates a temporary s_l using the ρ mutation strengths of the parental pool $\mathbf{E}_l$. This operator functions analogous to the recombination methods for the object variables, explained in Subsections 2.4.1 and 2.4.2, with the simplification that it recombines scalar quantities. The latter line introduced, L8, generates the mutation strength $\tilde{s}_l$, which will be used in the generation of the l-th descendant, using s_l and some external strategy parameters. L8 is explained in Subsection 2.5.1.

To summarize, the self-adaptation operator is introduced in Algorithm 5. If this operator is used, the mutation strength evolves during the loop of generations, simultaneous to the search for an optimum. The σ-SA operator, its type (see Subsection 2.5.1), and the σ-recombination rule used in L7 *must* be stated explicitly, since it cannot be given in the compact "(...)-ES" notation. Algorithm 5 can easily be adapted for the $(1, \lambda)$-ES and the (μ, λ)-ES algorithms. In both cases, L7 becomes obsolete.

2.5.1 Multiplicative σ-SA rules (MSR)

In L8, $\tilde{s}_l$ is generated using s_l and some exogenous parameters, by simply multiplying s_l with a pseudo-random number ξ

$$\tilde{s}_l := \xi \cdot s_l \quad . \tag{2.15}$$

There are several ways of generating ξ. Four of the proposed ones are listed here. More information can be found in [Sch95], [Fog95] (in a slightly different form), [Rec94], and [Bey96c, Bey96b], respectively.

$$\text{LogNormal} \qquad \xi := \exp(\tau\mathcal{N}) \tag{2.16}$$

$$\xi := 1 + \tau\mathcal{N} \tag{2.17}$$

$$\text{(Symmetric) TwoPoint} \qquad \xi := \begin{cases} 1/(1+\beta) & \text{if } \mathcal{U} < 1/2 \\ 1+\beta & \text{if } \mathcal{U} \geq 1/2 \end{cases} \tag{2.18}$$

$$\text{Generalized TwoPoint} \qquad \xi := \begin{cases} 1-\beta^- & \text{if } \mathcal{U} < p_- \\ 1+\beta^+ & \text{if } \mathcal{U} \geq p_- \end{cases} \tag{2.19}$$

In these equations, $\mathcal{N}$ stands for a pseudo-random sample of the standard normal distribution, as introduced in Section 2.1. Equation (2.17) is obtained as the first two terms of the Taylor expansion of (2.16). A pseudo-random sample of the uniform distribution with the range $[0, 1)$ is represented by $\mathcal{U}$. The other symbols $(\tau, \beta, \beta^-, \beta^+, p_-)$ are exogenous strategy parameters, i.e. the user has to specify them. The operation intervals can be given as

$$\beta^- > 0 \qquad \beta^+ > 0 \qquad 0 < p_- < 1 \qquad \beta > 0 \tag{2.20}$$

The values of these exogenous parameters affect the self-adaptation process of σ. The optimal values of them depend on μ, λ, ρ, N, on the ES algorithm used, and of course on the fitness function. For more information, the reader is referred to the related works mentioned in the chapter for State of Research, Chapter 5. For $p_- = 1/2$, $\beta^+ = \beta$, $\beta^- = \beta/(1+\beta)$, the GENERALIZED TWOPOINT rule reduces to the TWOPOINT rule.

2.6 Some possible alterations

In the previous sections, five standard ES algorithms are given exemplarily in their basic form. However, there are many other ways to implement each step of these algorithms. Some suggestions are mentioned here without a claim of completeness.

The ideas stated in this section as well as the five algorithms given previously are *not* canonical rules. The user can tailor them to the application needs. However, any alteration of the algorithms may alter their convergence properties.

The reader is referred to [SR95] for the contemporary state of the ES, and to [BFM97, Rec94, Sch95, Bey96c, Bäc96, Rud97] for further literature and thorough information.

Naturally, almost all alterations mentioned in this section can be used in combination; but they may make the theoretical analysis much more difficult.

2.6.1 L3: Where to start

At which state of the search space should an algorithm start, or what is the best location to start searching? This is a philosophical question which is hard to answer. The object variables for the optimum describe the best location to start; however, no search is necessary if it is known. If we do not know at all what we are looking for, or specifically, where the optimum is located, we can start anywhere: The starting point can be selected randomly. Generally, it is assumed in optimization that optimum variables (object variables for the optimum) are unknown.

If other search algorithms already delivered some results, these can be used in L3. Conversely, the results of the ES can be used as starting point by other algorithms.

It is practical to start at previously obtained variable settings with high fitness values, hoping to be near to optimum variables; it is also wise to start at a very different variable setting in each simulation run if the fitness function is multi-modal [Rec94, p. 155].

If the mutation strength is variable (as in Algorithm 5), it also has to be initialized in L3: In this case, each one of the $s_m^{(0)}$ values can be different. In general, if $s_m^{(0)}$ or $\sigma^{(0)}$ are chosen to be too large, the comma strategy may lose the good start position.

If the fitness function handles constraints, $\mathbf{P}^{(0)}$ must fulfill all of these. Lastly, the local performance of an ES can be measured *statically*. For this purpose, one determines the local conditions of interest and expresses these as variable settings on the search space. These settings are used as object variables of the individuals in $\mathbf{P}^{(0)}$, with the desired population distribution. L14 is removed from the algorithm, i.e. $\mathbf{P}^{(g+1)} := \mathbf{P}^{(g)}$. Therefore, the algorithm is repeated at the same initial conditions. This local performance can be measured in formal terms using convergence measures (see Chapter 4).

2.6.2 L4: Termination condition

The most trivial way to terminate an evolutionary search algorithm is to execute the loop of generations just for a fixed number of times. This number, G, gives the simulation length.

Other conditions may also be used, these are defined formally in [SR95]. Such conditions are mostly based on the quantity of, and/or relative/absolute change in the fitness values of (successive) populations, in their variable settings or strategy variables, and require the collection of simple statistics. They contain comparisons based on the best or worst individual in the population, or in the mean value of the respective quantity. The comparisons can be made over successive generations, or repeatedly after several generations; furthermore, different criteria can be used in the different phases of the simulation run, or the criterion itself can depend on time.

If the performances of other algorithms are known, these can be stated to serve as minimum requirements. For example, simulation length can be limited by the execution time, i.e. by the “wall-clock”.

2.6.3 L5: Variable λ

All of the five algorithms introduced in Section 2.1 through Section 2.5 produce λ offspring per generation throughout the whole simulation run.

Practically, the user can alter these algorithms to generate more or less offspring in each generation, based on his own criteria, e.g. based on the fitness function values. Therefore, λ becomes a variable which can be changed during the simulation. For instance, if the descendants created already yield a large quantitative improvement in the fitness space, one may stop creating further individuals in this generation, provided that μ individuals are available for the next generation. Therefore, less than λ descendants may be created in a single generation.

Handling constraints. Actually, more than λ descendants per generation may also be created. This is for example the case if constraints should be handled in addition to the fitness function evaluation. If an individual does not fulfill all constraints, it is called *lethal*; its fitness value is undefined, it cannot produce further individuals, and it should not be included in $\mathbf{P}^{(g+1)}$. Therefore, the loop L5-L12 is executed *until* λ *feasible* descendants are produced, i.e. indefinite number of times. Other methods also exist for constraint handling [BFM97, Chapter C5], [Deb98].

2.6.4 L6: Mating selection

In L6, ρ parental individuals for the recombination are selected. For the (μ, λ)-ES case, we have $\rho = 1$. Principally, the fitness values of the individuals in $\mathbf{P}^{(g)}$ are not considered in the ES at this step. However, one can define the probability to occur in $\mathbf{E}_l$ using the fitness values of the individuals in $\mathbf{P}^{(g)}$; the parents can be selected depending on their fitness values. As a result, individuals with higher fitness will occur more frequently in the parental pool. Consequently, the offspring are generated by the parents with better fitness values.

The determination of the mating pool $\mathbf{E}_l$ based on the fitness values introduces a computational overhead which may be small compared to the fitness evaluations. Only some selection schemes will briefly be mentioned here. Actually, these are used in algorithms closely related to ES, see Section 2.8 for literature references.

In *proportional selection*, each parent is assigned a selection probability proportional to its fitness. For this case, a scaling based on the fitness values, or on a function of fitness values, is necessary. That is, in order to be able to determine the proportion of each individual, all fitness values should be larger than zero, or they should be mapped to the interval $\mathbb{R}^+$ first. Actually, such a scaling can also be used even in the case when all fitness values are already positive; in order to fine-tune the proportions of individuals with fitness values above or below the average. For example, if a sufficiently *large* number is added to the fitness values, the proportional selection can be switched off: The case explained for the mating selection in Section 2.3, which is used as standard in ES, emerges as a special case of the proportional selection.

In *q-tournament selection*, $q \leq \mu$ individuals are randomly selected from $\mathbf{P}^{(g)}$ first. The one with the largest fitness joins $\mathbf{E}_l$. This tournament is repeated ρ times. If $q = \mu$ is chosen in this scheme, only the best parent generates offspring; therefore, μ effectively reduces to one.

The *linear ranking selection* scheme ranks the parents in $\mathbf{P}^{(g)}$ according to their fitness values. The probability to be selected for the mating pool depends neither directly nor absolutely on the fitness values, but on their ranking order instead.

The effect of such fitness-based schemes on the convergence behavior is not analyzed in this work. Such schemes are especially of interest in case of the $(\mu \overset{+}{,} \mu)$-type algorithms, which are also not analyzed here. Actually, combining them with truncation selection may be harmful, since it may cause an overemphasis of fitness values. Additional mating restrictions may be introduced, so that similar individuals may be encouraged (or discouraged) to recombine (see [BFM97] for more information).

2.6.5 L10: The mutation distribution

Several distributions can be used to generate mutations for real-valued variables, other than the normal distribution used in this work. The basic requirements on the mutation distribution are stated in [Bey96c, p.8]. They can be summarized as follows:

- **reachability:** Any state of the search space should be reachable from any other state in a finite number of generations.
- **scalability:** The distribution should be scalable by using the strategy parameter(s).
- **no bias:** The distribution should be chosen according to the maximum entropy principle.
- **symmetry:** The expected value of the mutation distribution should be zero.

These requirements are not very stringent. Several probability density functions are applied in the literature to generate mutations in the ES algorithm. For example, the spherically symmetric distribution is used in [Rud97]. In [Kap96], the performance of the Cauchy-distributed mutations are compared against the normally distributed ones. A more broad comparison can be found in [Müc89].

2.6.6 L10: The mutation vector/matrix

In Equation (2.2), the same mutation strength is used for the mutations on N variables. Therefore, using a single σ, an N-dimensional pseudo-random vector is generated. Principally, a different mutation strength can be used for each variable. In this case, one would have an N-dimensional σ *vector*, emphasizing the search in each direction with different mutation strengths. The mutation operator obtained in this way is aligned with the variable axes: Using a *covariance matrix* with non-zero off-diagonal entries, the emphasized directions can be directed freely in the N-dimensional space. A σ vector has up to N parameters, and a mutation matrix has up to $N(N+1)/2$ parameters; see [SR95] for details. Scaling the mutation strength differently in different directions will guide the search, or at least weight it in the preferred directions; although fulfilling the requirements stated in Subsection 2.6.5. This methods are expected to alter the convergence properties of ES.

2.6.7 L10: Other data types

The mutation operator should be adapted if the fitness function is only defined for the set of *integer* or *discrete* values. That is, the variable settings of the descendants should be in the same set as their parents.

The first way is to use a geometric distribution, which is similar in shape to the normal distribution, but has only integer values in its range. The second way is to use a normal distribution, and round the real numbers produced to the nearest integer. In case of discrete values, a mapping scheme should be introduced additionally to obtain the corresponding discrete value for any given real value. Please note that the set $\mathbb{B} = \{0, 1\}$ (*binary* values) is a special case of discrete sets, further examples are $\{A, C, G, T\}$, $\{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$, and $\{\boxminus, \boxtimes, \boxdot, \boxplus, \ominus, \circledast, \circledcirc\}$.

The search spaces with variables of different types (real, integer, and/or discrete) are called *mixed-integer*. For instance, one can construct a search space with five real values, seven natural numbers, and ten boolean variables, in short $\mathbb{R}^5 \times \mathbb{N}^7 \times \mathbb{B}^{10}$; the mutation distribution of each variable should accord the data type of the variable on which it is operating. Obviously, ES is not a method which is restricted to the search space $\mathbb{R}^N$.

2.6.8 L9: Other recombination operators

The intermediate and dominant recombination operators are introduced in Subsection 2.4.1 and Subsection 2.4.2, respectively. These can be extended to weight the parents. This weighting can be based on the fitness values of the parents, or on user-defined heuristics. It both cases, one assumes that the individuals with higher fitness values are nearer to the optimum than others. Further extensions which try to approximate the local topology and thereby make estimates for regions with higher fitness values are also possible, see [Rud97, Ost97, SV98]. All these extensions are expected to alter the performance of the standard ES in Section 2.4, in a good or bad sense.

For the dominant case, the weighting causes that some parents in $\mathbf{E}_l$ are selected more frequently than the others to determine the variable settings in $\mathbf{y}_l$ (see (2.13) and (2.14)). In the intermediate recombination, such weights cause that $\mathbf{y}_l$ gets nearer in the search space to the parents which are higher-weighted. Note that $\mathbf{y}_l$ is always placed in the subspace covered by $\mathbf{E}_l$. A recombination operator similar to the dominant recombination is used in the research field of genetic algorithms (see Section 2.8, cross-over, k-point cross-over).

2.6.9 L11: The evaluation of the offspring

Depending on the fitness function, the operation of L11 may differ from the case explained in Algorithm 1. Furthermore, some extensions may be necessary in the structure of the $\mathbf{P}^{(g)}$ for the realization. Some of these cases will be mentioned briefly in this subsection.

Variable N. For some technical optimization problems, the number of variables itself in the fitness function can be considered as *variable*. Introduction of further variables will increase the complexity of the technical constellation, reduction of these will simplify it. Therefore, N must be represented in the fitness function with a punishment term. The alteration of N can be realized in L10, causing a change in the structure of $\tilde{\mathbf{P}}^{(g)}$. The reader interested in an example where the gene deletion and duplication are realized is referred to the nozzle experiments [KS70].

Multi-criteria optimization, also called *vector optimization*. The fitness function may be expected to optimize multiple criteria simultaneously. If these cannot be weighted to give a scalar fitness value, then the ES algorithm must optimize a vector. Consequently, the $\tilde{F}_l$ values become vectors, and are principally not comparable beyond the limits of a Pareto set. For more information on multiple criteria decision making and optimization, see e.g. [HM79].

Moving optima. The fitness function may change over time. As a consequence, the fitness landscape changes. Therefore, the $\tilde{F}_l$ (and $F(\mathbf{y}_m^{(g)})$) values are only valid for the generation in which they are evaluated. If the plus selection strategy is used, it is advisable to reevaluate the parents since their fitness values may change drastically after several generations.

Noise-perturbed fitness. Actually, perfect and exact measurements do *not* exist in the nature. All measurements are perturbed with noise. Therefore, formal algorithms are required to function also under noise-perturbed noise for real world applications. The convergence behavior of ES under noise is investigated e.g. in [Rec94, Ch. 14], [Ott93], and [Bey93, Bey96c].

Polyploidy. An individual may have more than one set of variable settings (alleles). In nature, the existence of multiple sets of chromosomes in the genetic code (*genome*) of a single individual is called *polyploidy*. A single functional set of chromosomes is called *haploid*. Individuals having two such complete sets are called *diploid*. All of these three cases exist in nature. Polyploidy can be imitated in the optimization, e.g. for multi-criteria optimization. For the ES, it requires the redefinition of the fitness evaluation: The fitness value should be obtained based on multiple variable settings of the individual. This approach is already used in the literature for multi-criteria optimization (e.g. [Kur91]).

2.6.10 L14: Other selection strategies

The plus and comma selection strategies are introduced in Section 2.1 and Section 2.2, respectively. Others beyond these are also thinkable. The individuals may be allowed to live a fixed number of generations, κ, at most [SR95]. This strategy comprises the comma and plus strategies as the two extremes, for $\kappa = 1$ and $\kappa = \infty$, respectively. Additional to the variable settings, fitness value, and strategy parameters, the "age" of the individual must also be stored in the population. Some other possible alterations related to the selection operator are briefly mentioned below.

Variable μ. The number of parents may also be considered as variable. This makes the ES algorithm more complicated, and requires rules for determining $\mu^{(0)}$ and $\mu^{(g+1)}$ based on the $\tilde{\mathbf{P}}^{(g)}$. This extension, the variation of the selection pressure during the simulation, may be useful in some applications. For more details, see [Rec94, p. 96].

Best so far. By definition, the plus strategy always contains the best μ individuals obtained during g generations in $\mathbf{P}^{(g+1)}$. In contrast, the comma strategy moves through the search space without registering the individuals with good fitness values, and the individuals of the final population may have moderate fitness values when the simulation run ends. Therefore, the algorithm can be extended to store the best fitness values and corresponding variable settings in an external list. The maximum length and the structure of this list can be specified by the application programmer.

Time dependent selection. After the generation of λ descendants using $\mathbf{P}^{(g)}$, the individuals of the next generation are selected in the next step. One can define a probability distribution to generate pseudo-random values to be used by the selection scheme. This algorithm is a hybrid of the plus and comma selection strategies: A child better than its parent is selected always. Otherwise, it is selected if the randomly generated value is larger than a predefined threshold. This threshold can be changed during the simulation run so that the selection scheme is similar to the comma strategy at the beginning and to the plus strategy at the end. This scheme is adapted from simulated annealing (see Section 2.8).

2.6.11 L8: Other self-adaptation rules

In Section 2.5, the mutation distribution was assumed to be isotropic. Other mutation distributions were introduced in Subsection 2.6.6. If the strategy variables used to generate mutations constitute a vector, each entry in the vector can be adapted separately using the σ-SA rules. If we have a covariance matrix, the matrix generated by the σ-SA rule must obey the rules stated in Subsection 2.6.5. The principal axes of this matrix can be rotated using the matrix transformation rules. For more information on self-adaptation of the covariance matrix, see [Sch75, SR95]. Some further ideas will be briefly mentioned below.

Adapt sometimes. The σ-SA rule is applied in each generation as to Algorithm 5. Other algorithms can be designed which adapt the mutation distribution less frequently, such as every n-th generation, or based on some heuristics. Moreover, one may suggest *not* to alter the mutation distribution for each of the descendants generated: Some part of the offspring population may get the strategy parameters of the parents unaltered.

Additive σ-SA rules. Differing from the scheme in (2.15) of Subsection 2.5.1, one may suggest σ-SA rules of type

$$\tilde{s}_l := s_l + \xi, \qquad \mathrm{E}\{\xi\} \approx 0, \qquad \mathrm{P}(\xi > 0) \approx \frac{1}{2} \ , \tag{2.21}$$

i.e. with the expected value $\mathrm{E}\{\tilde{s}_l\} \approx s_l$ and with the same probability for increasing or decreasing the mutation strength. Such σ-SA rules are called additive, and are not investigated in the literature.

The 1/5-th *success rule.* The control of the mutation strength σ is actually part of the original $(1+1)$-ES algorithm [Rec65]. It was not given in the algorithm in Section 2.1 in order to present the algorithm in its simplest form. The creation of a descendant with a fitness value better than its parent is considered as *success*. The "*one-fifth*" rule is an external control mechanism based on the fitness values of the descendants, which tries to hold the proportion of the successful descendants to the total number of individuals generated close to the ratio 1/5. For this purpose, one collects the statistics for the successful and unsuccessful individuals. In user-defined periods, the ratio of the number of successful descendants to the total number of offspring generated is calculated. If this measured ratio is larger than 1/5, than σ is increased, otherwise σ is decreased. The optimal ratio for maximal progress toward optimum depends on the fitness function, and on the local state in the search space. How much σ should be changed with respect to the parental value requires another heuristic.

Derandomization of σ-SA rules. The rules in Subsection 2.5.1 use pseudo-random numbers for σ-SA. There exist other methods for σ-SA *without* using random numbers. The interested reader is referred to [Ost97].

2.6.12 L7: Recombining strategy parameters

We limited the strategy parameters to the parameters used by the mutation distribution, although one could think of further uses for these. If the recombination operator is used together with the self-adaptation operator, the strategy parameters of a descendant are generated using the ones of its parents. The case for isotropic mutations is explained in Section 2.5. If the extensions introduced in Subsection 2.6.6, i.e. the mutation vector and the covariance matrix, are used, the same steps for recombination should be repeated separately for each member of the strategy parameter set. Any weighting scheme analogous to the ones introduced in Subsection 2.6.8 is also plausible. Additional to these, the geometrical mean of the parental σ values ($\sqrt[\rho]{s_1 \cdot s_2 \cdots s_\rho}$) can also be used in the generation of the offspring [Ost97].

2.7 The hierarchical ES

Since resources like computation time or memory space are not as scarce as in the days the $(1+1)$-ES was first designed [Rec65], other more complex ES algorithms are suggested and applied in the literature [Rec78, Her92].

The basic idea behind such algorithms is the realization of parallel sub-populations. In this way, populations with different μ, λ, or ρ values, or with different strategy parameters can be executed simultaneously. As a result, the performance obtained for different algorithms can be used to optimize the *parameters* of the ES algorithm (e.g. μ, λ, ρ, mutation distribution, etc.). In other words, different ES algorithms run in the first level; and in the second level we have *another* ES algorithm which evaluates the outputs of these algorithms.

Using this information, it chooses and/or generates the algorithms to be used in the next "algorithm generation".

The first subsection supplies some preliminary information on the time and space complexity. The second subsection gives some examples for the hierarchical ES, extracted from [Rec94, pp.81-100].

The hierarchical ES is also called *nested* ES in the literature [BFM97]. Actually, it is also denoted as *Meta*-ES, since we evaluate the populations in the second level and individuals of these populations in the first level.

Some examples for hierarchical ES algorithms will be given below. Only the two-level hierarchical ES will be mentioned here, although algorithms with more levels are thinkable (e.g. [Her92]). Any of the ES algorithms mentioned in the first five sections can be used in the first level. The functioning mechanism of the second level will be explained by examples.

The simulation length of the first and second level will be indicated in the concise ES notation in this subsection, as γ and γ', respectively. For example, $(\mu/\rho_{\mathrm{I}}, \lambda)^{\gamma}$-ES compactly denotes the first level. γ is also called the isolation time or *isolation period* of the $(\ldots)$-ES concerned. Of course, if the σ-SA operator is applied, its type (L8, Subsection 2.5.1) and the sort of recombination used in L7 must be stated explicitly.

It is already stated that the second level works on the level of populations, and it uses complete populations (or, as a special case, a single individual) in its operators. There are many ways to do this, and some of these will be explained here using examples.

The $[\mathbf{3}, 5(\mathbf{4}, 7)^{30}]^{10}$-ES. This notation describes that we have a $(\mu, \lambda)^{\gamma}$-ES running on the first (or lower) level, with $\mu = \mathbf{4}$, $\lambda = 7$, for $\gamma = 30$ generations each. The second level has initially **3** populations, which are selected and duplicated to give 5 populations. For this duplication, the parent populations are selected analogous to (2.4). How one can design a mutation operator on the population level is open for discussion. Each of these populations have **4** individuals, and perform the $(\mathbf{4}, 7)^{30}$-ES algorithm for the isolation length of 30 generations. Thereafter, the *average* fitness values of these 5 populations are computed, and the populations having the best **3** values are selected as the next generation in the second level. The second level is executed $\gamma' = 10$ times. Actually, γ' can be omitted in the compact notation, as one must not explicitly state G in a single-level ES. The number of parental populations and the number of parental individuals per population are indicated here in bold face.

The $[\mu' \overset{+}{,} \lambda'(\mu \overset{+}{,} \lambda)^{\gamma}]^{\gamma'}$-ES. This one is the general case of the previous example. Please note that we have μ' *parental* populations of μ individuals each. Instead of this algorithm, one might prefer $(\mu' \cdot \mu \overset{+}{,} \lambda' \cdot \lambda)^{\gamma' \gamma}$-ES. However, the essential power of hierarchical ES is that one can use different strategy parameters in the isolated populations. The populations operating with different mutation strengths are expected to yield different average fitness values at the end of the isolation period γ.

The $[\mu^{'}/\rho^{'} \overset{+}{,} \lambda^{'}(\mu/\rho \overset{+}{,} \lambda)^{\gamma}]^{\gamma^{'}}$**-ES.** In this example, the first level is the $(\mu/\rho \overset{+}{,} \lambda)$-ES explained in Algorithm 4, with the isolation time γ. Also Algorithm 5 (with self-adaptation) can be used here as well. Please note $\lambda^{'} > \mu^{'}$ and $\lambda > \mu$ if the comma selection strategy is used. The recombination operator on the second level is introduced in this example. It is analogous to the dominant recombination operator explained in Subsection 2.4.2. In the second level, the dominant recombination produces populations using the complete *individuals* of the parental populations, as it uses complete variables in the first level to produce individuals. The number of populations recombined per population produced is given by $\rho^{'}$, $\rho^{'} \leq \mu^{'}$.

The $[\mathbf{2}, (\mathbf{1}, 5)^{20}\,;\,(\mathbf{1}, 10)^{10}\,;\,(\mathbf{1}, 20)^{5}]^{30}$**-ES.** Here, the first level consists of three different algorithms: The $(\mathbf{1}, 5)^{20}$-ES, the $(\mathbf{1}, 10)^{10}$-ES, and the $(\mathbf{1}, 20)^{5}$-ES. These three algorithms are expected to consume roughly equal amount of computation time. After the isolation period, the best two of three resulting individuals (since $\mu = 1$) are selected for the next generation. The second level is executed $\gamma^{'} = 30$ times.

Note that ";" is used here to separate the algorithms of the first level, instead of the "+" symbol proposed in [Rec94, p. 98]: The aim is not to overload the "+" symbol, which also represents the plus selection strategy.

$(\mu \overset{+}{,} \lambda_1\,;\,\lambda_2)$**-ES.** This notation is invented to express that λ_1 and λ_2 are generated using different mechanisms, e.g. different mutation distribution. It can be considered as the $[\mathbf{1}, (\mu \overset{+}{,} \lambda_1)^{1}\,;\,(\mu \overset{+}{,} \lambda_2)^{1}]$-ES.

Further ideas. One can design some heuristics for generating other, more promising strategy parameter values for the next generation in the second level, based on the average fitness values supplied by the first level after the isolation period γ. If different ES algorithms are running in parallel, similar heuristics can be designed which should suggest more promising algorithm parameters (such as μ, λ, ρ, etc.). Based on the performance of different algorithms in the previous isolation period, these parameters can be adjusted before the next isolation period.

2.8 Related algorithms

This chapter introduced some typical ES algorithms. Thereafter, it gave an overview of the extensions, that can be made on these five fundamental algorithms; some of these have already been realized. In Subsection 2.7, the hierarchical ES was introduced. This subsection will briefly mention some related evolutionary optimization techniques.

Evolution strategy (ES) belongs with *evolutionary programming* (EP), *genetic algorithms* (GA), and *genetic programming* (GP) to the class of *evolutionary algorithms* (EA). The research area of EA is also named as *evolutionary computation* (EC). The EA, *artificial neural networks* (ANN), and *fuzzy logic* (FL) are branches of *computational intelligence*

(CI). These relations can be summarized as

$$\begin{aligned} \mathrm{CI} &:= \mathrm{EA} + \mathrm{ANN} + \mathrm{FL} \\ \mathrm{EC} \equiv \mathrm{EA} &:= \mathrm{ES} + \mathrm{EP} + \mathrm{GA} + \mathrm{GP} \ . \end{aligned}$$

The similarities and differences between the three primary members of the EA (ES, EP, and GA) can be read in [BS93]. The genetic algorithm originated in the early Sixties from the modeling of general adaptive processes [Hol75]. It is later applied to a rich number of domains [Gol89]. Evolutionary programming was constructed first in the early Sixties by L. J. Fogel for the evolution of finite state machines [FOW66]. For the current state of EP, the reader is referred to [Fog92]. Evolution strategies were first used for discrete experimental optimization tasks [Rec65], but later also to real-valued and mixed-integer problems on computers. For a brief history of the ES, see the chapter devoted to State of Research (Chapter 5).

Nowadays, all these three subclasses of EA are used primarily for optimization. The differences between them become vague. However, some of these differences, mostly historical ones, will be mentioned here briefly.

The operators of canonical GA work on the binary coding of the variables; however, there exist other GA versions that do not code the variable settings at all. Whereas ES directly uses the fitness values, EP and GA make use of a scaling. The self-adaptation operator is applied in ES and modern EP; principally, the mutation *rate* is kept constant in GA. For the GA community, mutation is originally assumed to be a background operator of negligible importance, emphasizing the importance of recombination (called also *cross-over*). However, for EP, mutation is the *only* variation operator, since the recombination is *not* used at all in this subclass. The EP models the evolution at the level of species, as a competition between the old and new species [Fog92]. In general, we know that species do not recombine. The modern EP uses a rule similar to (2.17) for the self-adaptation of the mutation strength. In contrast, the recombination operator is generally considered as the main variation operator in the GA community, and numerous methods exist to perform that. Both mutation and recombination are considered as being equally important for ES.

Several selection methods were mentioned in Subsection 2.6.4. A characteristic difference between the EA exists on the selection of the mating pool (L6). This view to the selection operation imitates the *sexual selection*, the selection of the parents for the generation of a descendant. Originally, GA used proportional selection; whereas ES used uniform selection and did not emphasize any parent by their fitness function values. However, these original schemes are not restrictive. Traditionally, EP uses repeated q-tournament selection in L14; actually, it can also be used in L6, as proposed in Subsection 2.6.4. GA uses all the offspring generated, unless the elitist selection is used. The truncation selection obviously needs a birth surplus. It is applied in L14, and imitates the population reduction caused by environmental effects. This could also be imitated in GA; however, any research in this direction is unknown to the author.

The cross-over operator used in GA differs from the intermediate and dominant recombination in ES, although being similar to the latter. In GA, the number of parents selected for the cross-over operation is traditionally two, although $\rho > 2$ yields very good results [ERR94]. The cross-over may be realized anywhere on the bit-strings of the parents, but at the same position(s) on both parental bit-strings. One or both of the bit-strings produced may be used, depending on the implementation. The number of cross-over points (one-, two-, or k-point *cross-over*) as well as the cross-over probability are also implementation dependent.

The mutation operators of modern EP and ES both use normal distributions (Equations (2.1) and (2.2)), and are very similar to each other. In GA, the mutations applied invert the bits of the coding: The expected value of the number of bits inverted per individual depends on the *mutation rate*, given by the user. The bit inversions can occur anywhere, they are distributed uniformly. As a result, small changes and large changes on the variable settings are equally likely. For the GA without coding (also called real-coded GA if operating on real values only), other mutation schemes are also conceivable which generate small changes more frequently than large ones.

As one can see, it makes not much sense to discuss which specific algorithm should be applied. Such discussions are becoming more and more obsolete. Actually, it is essential to ask *which* operators are appropriate for a specific fitness function, or which specific *type* of an operator gives better results than others. A specific algorithmic approach may be more appropriate than the others for a given problem class.

Genetic programming. The fourth member of EA is GP. It developed from the GA approach, and can be formulated as a struggle for finding the optimal computer program for a given task. It *is* a member of EA since the programs of *variable* length are considered as individuals, and the search process can be seen as the evolutionary loop L4-L17. Mostly, these programs are represented in LISP code, or in corresponding trees [Koz94]. However, this is not a must. For example, the same evolution process is realized in [Nor97] using machine code, considering the set of instructions as a linear list. The book [BNKF98] is another good reference to GP.

The breeder genetic algorithm (BGA), introduced in 1992, lies between ES and GA. It uses the truncation selection in L14 as in ES, but emphasizes the importance of the recombination operator and considers the mutation as a background operator. The mutation *rate* used is inversely proportional to N. Therefore, only one variable is mutated at a time on the average. However, there are BGA variants for which this conclusion is not true [VMC95].

Simulated annealing. This iterative search method imitates the cooling process of liquid materials [KGV83]. It is normally not considered as a member of EA; however, it can be seen as the (1+1)-ES, with a constant mutation strength and stochastic time-dependent selection operator. The descendant always substitutes its parent if it has a better fitness value. If its fitness is worse, it substitutes its parent based on a heuristic: A pseudo-random number is generated using a probability distribution. If this number is larger than a threshold, the offspring substitutes its parent. The acceptance probability for this latter case decreases over time, since the threshold is increased over generations based on another heuristic. This scheme allows random movements in the search space at the beginning of the simulation run, and it becomes more and more similar to the $(1+1)$-ES in the later generations (see also Subsection 2.6.10; time dependent selection). The massively parallel SA is compared to the $(1+1)$-ES in [Rud94]. There are also simulated annealing variants that possess an external control mechanism for the mutation strength.

Tabu search. If somebody tries to find the optimum of a multi-modal fitness function, he has to overcome the local optimality traps. Tabu Search (TS) is designed for this purpose [Glo86, Glo89a, Glo89b, GTdW91]: It categorizes the old trials as "good" or "bad" based on their relative fitness values, and stores these in three levels of memory (short-term, intermediate term, and long-term). The good trials indicate candidate *aspiration* regions to be searched in the future (in the *diversification* phase); whereas the bad ones mark the opposite regions yielding an *intensification* caused by some *tabu* restrictions.

The generation of new trials is named as *aggressive exploration* in TS. This phase is the most essential part of the algorithm, it is guided by the short-term memory. The intensification and diversification modes use intermediate term and long-term memories, respectively, and guide where the search will continue. The tabu restrictions (criteria) and the aspiration regions should be updated continuously during the search since the categorization as good or bad is relative,

A recent book [GL97] is suggested for the interested reader; unfortunately, TS is not in the scope of this work. As a *meta-level* algorithm, it can be combined with other methods. It definitely has many interesting application areas as can be seen in the literature, mostly on integer and discrete domains.

Other methods. This section does not serve as a complete list of algorithms which are related to ES. Many other approaches to the search for the optimum exist. The book [Sch95] is advised for an overview of the direct (numerical) optimization methods. These methods are superior for some special fitness functions.

Chapter 3

Fitness functions

This chapter is dedicated to the fitness functions analyzed in this work. In addition, some related functions are mentioned. The fitness function is supplied by the user to the ES algorithm. It is used for the evaluation of the individuals in the lines L3 and L11. Only the fitness values obtained hereby are used by the selection operator (L14). The details can be found in the previous chapter, Section 2.3.

This chapter has three subsections. The first section supplies an overview of some mathematical terms, which can be considered as common knowledge. Thus, Section 3.1 can be skipped by the informed reader. The second section gives the core concepts on optimization relevant to this work. Finally, Section 3.3 introduces the fitness functions used in the analysis of the performance of the algorithms in Chapter 2. Furthermore, this last section presents some few other functions related.

3.1 Background on functions

This work is concerned with functions of the type

$$F:\ \mathbf{x} \mapsto F(\mathbf{x}), \qquad F:\ \mathbb{R}^N \to \mathbb{R}, \qquad N \in \mathbb{N}, \tag{3.1}$$

that return a scalar real value for each arbitrary input vector $\mathbf{x}$. The vector $\mathbf{x}$ has N real-valued components that represent a point in the N-dimensional *domain* of F. The value $F(\mathbf{x})$ is in the *range* of F, which is the set of real numbers in this work. $\mathbb{R}$ denotes the set of real numbers, and $\mathbb{N}$ the set of natural numbers (without zero), respectively. For functions with a domain other than $\mathbb{R}^N$, see Subsection 2.6.7; for a range other than $\mathbb{R}$, see Subsection 2.6.9.

This quite informal section briefly mentions some notions of the *calculus*. These terms are introduced in any regular freshman course in mathematics, e.g. [TF96]. Therefore, they are assumed to be a part of the common knowledge of university students and graduates. This section serves as a smooth refreshment, and Section 3.2 will be based on these terms. The important terms mentioned are given in *italic* so that the reader informed in these basic terms can just scan over this section.

As a demonstrative example, consider

$$y := F(\mathbf{x}) = ax_1 + bx_2 + cx_3 + d, \qquad a, b, c, d \in \mathbb{R};\ \mathbf{x} = (x_1, x_2, x_3)^{\mathrm{T}}, \tag{3.2}$$

with the *constants* a, b, c, and d; and with three (*independent*) *variables* x_1, x_2, and x_3. The function value y is also called the *dependent* variable. Using this function, the notion of monotonicity will be introduced. For instance, if we have $a > 0$ in (3.2), then $F(\mathbf{x})$ is monotonically *increasing* in x_1: That is, if all other variables are constant, any increase in x_1 will definitely give a larger $F(\mathbf{x})$ value. Analogously, $a < 0$ causes $F(\mathbf{x})$ to be *monotonically decreasing* in x_1. The special case $a = 0$ causes $F(\mathbf{x})$ to be *independent* of x_1.

These terms on monotonicity can be used to define maximum and minimum of a function. A change from being monotonically increasing to monotonically decreasing with respect to all independent variables locates a *maximum*, the reverse way of change a *minimum*. A formal definition is based on first and second order partial derivatives. This definition, along with the definitions of *smooth*, *discontinuous*, *continuously differentiable*, *one-to-one*, and *onto*, can be found in a regular calculus textbook.

The maximum with the largest function value is called the *global maximum*, all other maxima are called *local*. If two or more vectors in the domain of a function yield the value of the global maximum, then the global maximum is called *degenerate*. The global and local minima are defined analogously.

In the domain of a function, the independent variables can be restricted by some *end points* or *limits*. Such restrictions make the intervals *closed*, in their absence one has an *open interval*. For example, $F(x) = \log(x)$ has a domain $x > 0$, which is bounded in only one direction.

The variables describing the domain of a function are called *object variables*, implying that our objective is finding the maximum values of these. The term object variable is preferred in this work to the terms "search space variable" and "decision variable". The N variables span an N dimensional *space*. The *distance* between any two points in this space can be given by different measures. The *Euclidean* distance measure is of concern in this work. The term *vicinity* will be used in this work with the meaning "the immediate neighborhood of a point, where the properties of the quantity observed (mostly $F(\mathbf{x})$) does not change or changes very slightly".

3.2 Measuring the fitness value

In this section, some relevant terms in optimization will be introduced based on the basic terms of calculus mentioned in the previous section. In this way, the shift in the terminology will be represented. In the jargon of evolutionary algorithms, some notions of the calculus are renamed in order to stress the analogy to the selection process in biology. For example, the term *search space* is used instead of the domain of the fitness function, and *fitness value* instead of the dependent variable. Similarly, some other terms in optimization are equivalent to the respective notions in the calculus, as to be described next.

In general, **optimum** means [Yer96, p. 1011]

1. the best or most favorable point, degree, amount, etc. as of temperature, light, and moisture for the growth or reproduction of an organism.

2. the greatest degree or best result obtained or obtainable under specific conditions.

3. (adjective) best or most favorable.

The first meaning of optimum leads us to the field called multi-criteria optimization (see Subsection 2.6.9). This work will be restricted to the second meaning of optimum.

The process of searching for the optimum is called optimization, and the optimum can be defined either as the global maximum or as the global minimum. The search for the maximum of a fitness function is called *maximization*; and analogously, *minimization* is the search for the minimum. The ES algorithms try to find the optimum of a given fitness function.

Without loss of generality, the case of maximization is considered in this work. Therefore, the global maximum is called global optimum or just optimum. The local maxima will be named as local optima. Formally, the optimum will be denoted by $\hat{F}$ in this work, and the object variable vector of the optimum by $\hat{\mathbf{x}}$,

$$\forall \mathbf{x} \in \mathrm{I\!R}^N : \quad \hat{F} := F(\hat{\mathbf{x}}) \quad \text{such that} \quad F(\hat{\mathbf{x}}) > F(\mathbf{x}) \quad \text{if} \;\; \mathbf{x} \neq \hat{\mathbf{x}} \tag{3.3}$$

where $F(\mathbf{x})$ is defined in (3.1). If the optimum is *degenerate*, we have the same $\hat{F}$ value for different $\hat{\mathbf{x}}$ vectors (plural: *optima*). If $F(\mathbf{x})$ has several local optima, the search space is called *multi-modal*, otherwise *unimodal*. Please note that (3.3) is only valid for unimodal functions.

The term "fitness" should actually be formalized in order to use it unambiguously. The dictionary [Yer96, p. 537] gives 23 different explanations to "fit". In this work, "fitness" is concerned with the function F that is used to evaluate the individuals. The function F itself is called *fitness function*. Other equivalent names for it are objective function or quality function. The *fitness space* means nothing more than its range. The *fitness (value)* is just a member of this range ($\mathrm{I\!R}$). In (2.9), the fitness values of parental and offspring populations of the $(\mu + \lambda)$-ES are sorted in increasing order. This equation is repeated here for the reader's convenience, where $\gamma := \mu + \lambda$:

$$F_{1:\gamma} \leq F_{2:\gamma} \leq \ldots \leq F_{\gamma-1:\gamma} \leq F_{\gamma:\gamma} \; . \tag{3.4}$$

The *best* (or *fittest*) individual has the fitness value $F_{\gamma:\gamma}$. Similarly, $F_{\gamma:\gamma}$ is the *best* fitness value in this set of γ individuals. The individual with $F_{\gamma-k:\gamma}$ is *fitter* than the one with $F_{\gamma-l:\gamma}$ if $F_{\gamma-k:\gamma} > F_{\gamma-l:\gamma}$ ($1 \leq k < \gamma - 1$, $1 \leq l < \gamma$; of course, $l > k$). The *m-th fittest* (or *m-th best*) individual has the fitness value $F_{\gamma-m+1:\gamma}$ ($0 \leq m \leq \gamma$); the special case $m = \gamma$ gives the fitness value of the *worst* individual, the *worst fitness*. Note that if the probability distribution generating the fitness values of the offspring population is known, one can predict all $F_{\gamma-m+1:\gamma}$ values. The study discipline is called order statistics [ABN92].

However, the detailed knowledge is not a prerequisite for the comprehension of this work. The order statistics relevant to this work will be derived where necessary.

Lastly, some more terms on the constraints applied to the fitness function will be given. Formally, additional restrictions stated by the user on the domain are called *constraints*. The variable setting $\mathbf{x}$ fulfilling all specified constraints is called a *feasible* solution. Accordingly, the constraints are *satisfied* if they are not *violated*; and a satisfied constraint can be *active* or *inactive*.

From the viewpoint of the implementation, the fitness value given by $F(\mathbf{x})$ becomes invalid even if a single constraint is violated by $\mathbf{x}$. Such an individual is not directly usable (in scope of this work). Its fitness value must be altered accordingly: It is either reassigned to an arbitrary value smaller than all fitness values obtainable from the search space, or simply to $-\infty$ in order to indicate constraint violation.

3.3 Fitness functions of interest

In this section, the fitness functions of interest in the scope of this work are introduced. The case of maximization is considered in this work without loss of generality. The first subsection is devoted to the *sphere model* function. It has a great importance in the theoretical analysis of ES. Its relevance to this work is established in the next subsection (Subsection 3.3.2). This work is dedicated to the theoretical analysis of *ridge functions*, introduced in the second subsection. In the third subsection, the *corridor models* are introduced to the reader. This work will also investigate whether these models serve as asymptotic limit cases of ridge functions. Lastly, a one-dimensional polynomial function is introduced. It will be used in the visualization of the convergence measures (see Chapter 4).

3.3.1 The sphere model

The sphere model is the fitness function that is mostly used as the fitness function in the theoretical analysis of the performance of ES. In this model, the fitness values are distributed in a sphere-symmetrical manner around the optimum [Rec73, p. 115]. In Figure 3.1, a contour plot of such a model is given. The curves in the figure connect the states of the search space with equal fitness value. These curves will be named as *isofitness* curves (equivalent to isoquality or isometric curves; or isohypses). For $N = 3$, they become isofitness surfaces; for $N > 3$ isofitness hyper-surfaces. In this case, the isofitness hyper-surfaces compose concentric hyper-spherical shells. The optimum is located at the center of these shells. The fitness value increases monotonically as the distance D to this center decreases. Therefore, the sphere model can be formalized using a one-dimensional, monotonically increasing function W of the distance D to the optimum, for $D \geq 0$. Figure 3.2 exemplarily presents some members of the sphere model family.

The general case of the sphere model is given in (3.5), as $F_s(\mathbf{x})$. The Euclidean distance D to the optimum is defined in (3.6); and $F_s(\mathbf{x})$ is redefined using D. Several members of the sphere model are stated in (3.7), (3.8), (3.9), and (3.10). The contour

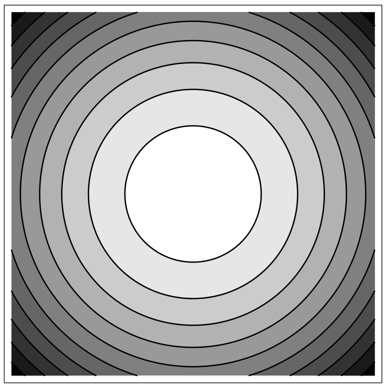

Figure 3.1: The contour plot of the sphere model for $N = 2$. $F(\mathbf{x}) = -x_1^2 - x_2^2$ is plotted here in the interval $\{x_1 \in \{-100, 100\}, x_2 \in \{-100, 100\}\}$. The concentric shells connect the sphere-symmetric states with equal fitness value, $\hat{\mathbf{x}} = (0,\, 0)^{\mathrm{T}}$. The brighter areas have higher fitness values. The axis labeling is omitted since it depends on $\hat{\mathbf{x}}$ and the function $W(.)$ chosen (see Equation 3.6).

$$F_s(\mathbf{x}) \equiv F_1(\mathbf{x}) := -W\left(\sqrt{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}\right) \tag{3.5}$$

$$F_s(\mathbf{x}) := -W(D), \qquad D := \sqrt{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2} \tag{3.6}$$

$$F_2(\mathbf{x}) := -D^2 \tag{3.7}$$

$$F_3(\mathbf{x}) := -\sum_{j=0}^{\infty} a_j D^{b_j} \tag{3.8}$$

$$F_4(\mathbf{x}) := -\sum_{k=0}^{\infty} c_k \exp(d_k D) \tag{3.9}$$

$$F_5(\mathbf{x}) := F_3(\mathbf{x}) + F_4(\mathbf{x}) \tag{3.10}$$

Figure 3.2: The *sphere model* is introduced in (3.5). In Equation (3.6), D denotes the distance to the optimum, $\hat{x}_i$ denote the components of object variable vector $\hat{\mathbf{x}}$ for the optimum. The optimum $\hat{F} := F_s(\hat{\mathbf{x}})$ depends on the monotonically increasing function W used. Additionally, some concrete members of the sphere model family are given ($F_2(\mathbf{x})$, $F_3(\mathbf{x})$, $F_4(\mathbf{x})$, and $F_5(\mathbf{x})$). $N \in \mathbb{N}$, $x_i \in \mathbb{R}$, $\mathbf{x} \in \mathbb{R}^N$, a_j, b_j, c_k, $d_k \in \mathbb{R}_0^+$.

plots of these for $N=2$ are similar to Figure 3.1; however, the fitness values for the same D will depend on the function used. $F_5(\mathbf{x})$ is the sum of $F_3(\mathbf{x})$ and $F_4(\mathbf{x})$, and $F_2(\mathbf{x})$ is a special case of $F_3(\mathbf{x})$. The symbol $\mathbb{R}_0^+$ represents the set of nonnegative real numbers, i.e. $a_j \in \mathbb{R}_0^+$ means "$a_j \in \mathbb{R} \wedge a_j \geq 0$".

The sphere model establishes the framework for the analysis of ridge functions. For $\hat{\mathbf{x}} = \mathbf{0}$, the similarity will become more clear. Note that the set of symbols used here differ from the ones used in the original work of Rechenberg. The symbols Q, r and R will represent other quantities in this work.

3.3.2 The family of ridge functions

The family of ridge functions will be introduced in this subsection. The overview of this function family is given in Figure 3.3. Equation (3.11) states the general case of ridge

$$F_R(\mathbf{x}) \equiv F_6(\mathbf{x}) := x_0 - d\left[\sum_{i=1}^{N-1} x_i^2\right]^{\frac{\alpha}{2}} \tag{3.11}$$

$$r := \sqrt{\sum_{i=1}^{N-1} x_i^2} \tag{3.12}$$

$$F_R(\mathbf{x}) := x_0 - dr^\alpha \tag{3.13}$$

$$F_7(\mathbf{x}) := x_0 - d \tag{3.14}$$

$$F_8(\mathbf{x}) := x_0 - dr \tag{3.15}$$

$$F_9(\mathbf{x}) := x_0 - dr^2 \tag{3.16}$$

$$F_{RR}(\mathbf{x}) \equiv F_{10}(\mathbf{x}) := \mathbf{v}^{\mathrm{T}}\mathbf{x} - d[\|(\mathbf{v}^{\mathrm{T}}\mathbf{x})\mathbf{v} - \mathbf{x}\|]^\alpha \tag{3.17}$$

Figure 3.3: The family of ridge functions is presented here. The general case of ridge functions is given in (3.11). The distance to the ridge axis, r, is defined in (3.12), yielding (3.13) from (3.11). Equation (3.14) is obtained for $\alpha = 0$, (3.15) for $\alpha = 1$, and (3.16) for $\alpha = 2$. These three cases are called hyperplane, sharp ridge, and parabolic ridge, respectively. The last equation, (3.17), represents $F_{RR}(\mathbf{x})$, the rotated form of the general ridge function with the unit vector $\mathbf{v}$ in the ridge axis direction. $N \in \mathbb{N}$, $\alpha \in \mathbb{R}$, $d \in \mathbb{R}_0^+$, $\mathbf{v} \in \mathbb{R}^N$, $\|\mathbf{v}\| = 1$.

functions. The object variable vector for the optimum of this *unimodal* function for $\alpha > 0$ reads

$$\hat{x}_0 \to +\infty, \qquad \forall i \neq 0 : \ \hat{x}_i = 0 \ . \tag{3.18}$$

Since (3.18) lies outside the definition interval, one can say that ridge functions do not have any optimum. However, one also can take another point of view by considering ridge functions as a limit case. The aim is to obtain (3.13) as a limit case of another function.

The following function ($a \in \mathbb{R}$, $a < 0$, see (3.12) for the definition of r) may serve this purpose

$$F(x_0, r) \;:=\; x_0 + a x_0^2 - d r^{\alpha} \;. \tag{3.19}$$

It has its maximum at ($\hat{x}_0 = -1/2a$, $\hat{r} = 0$). As $a \to 0$, the optimal x_0 gets the limit value $\hat{x}_0 \to \infty$ (remember that a is negative). At this limit, one obtains the function

$$F(x_0, r) \;:=\; x_0 - d r^{\alpha} \tag{3.20}$$

which is nothing but the general ridge function in (3.13). Therefore, the optimum of ridge functions will be considered as a limit case in the following. For $F_{RR}(\mathbf{x})$ in (3.17), $\hat{\mathbf{x}}$ depends on $\mathbf{v}$. The object vector is denoted as $\mathbf{x} = (x_0, x_1, \ldots, x_{N-1})^{\mathrm{T}}$ in order to underline the specialty of the first variable. Any movement in the search space causing an increase in the object variable x_0 is privileged linearly. The search for the optimum is unbounded in this direction. Hence, the subspace having relatively better fitness values around any state in the search space is unbounded. This subspace defined by relatively fitter neighboring states is called the *success region.* The success region is even larger for $\alpha \leq 0$. For the $\alpha = 0$ case, the $\hat{x}_i$ values do not appear in the definition of the object variable vector of the optimum (c. f. Equation (3.18)).

However, as shown in (3.18), the success region is bounded with respect to other x_i for $\alpha > 0$. This becomes more clear when the variable r introduced in (3.12) is used in the function definition: r can only be decreased down to zero, but not further. Additionally, the minimization of r should be accomplished simultaneous to the maximization of x_0. Since the ES algorithm does not have any internal information about the fitness function, these subgoals of maximization and minimization must be accomplished together and simultaneously, weighted by d and α.

One highly significant property of ridge functions becomes more clear if we express this observation in other words: The maximization subgoal is unbounded, and the parameters d (multiplicative weight) and α (exponential weight) can be tuned to influence the effect of the object variables in r on the fitness value. Therefore, the maximization of x_0 can be made more difficult than the minimization of r, since only the scalar fitness values are available to ES. Consequently, the maximization of x_0 becomes automatically an implicit *long term goal.* Moreover, with its various values for d and α, the simply-structured family of ridge functions serves us as a simple and *scalable* test-bed of functions for optimization. One should expect a change in the convergence behavior of the ES on the ridge functions depending on the values chosen for d and α.

As already mentioned, the minimization of r in (3.13) is the second subgoal for the maximization of $F_R(\mathbf{x})$. Especially for $\alpha \gtrapprox 2$, or for smaller α values ($\alpha \gtrapprox 1$) with large d ($d \gg 1$), a change in the value of a variable in r makes a larger effect on the fitness value than the same quantitative change in x_0. For the same Euclidean distance traveled, a change in r will cause a greater effect on the fitness value than the change in x_0. Since the selection in line L14 of the ES algorithm is based on the fitness values, this expectation should be observable in the behavior of the algorithm over generations. Consequently, the minimization of r will be named as an implicit *short term goal.*

The important members of the ridge family are obtained for $\alpha \in \{0, 1, 2\}$. These three functions are named as *hyperplane*, *sharp ridge*, and *parabolic ridge*, respectively. They will be introduced next, using contour plots of the latter two. Thereafter, the general rotated ridge function will be introduced. Its interpretation establishes another reason for investigating ridge functions. The explanation of two important properties of ridge functions concludes this subsection: The relation of the ridge functions to the sphere model will be explained, and the case for large α values will be shortly mentioned.

The hyperplane. The *hyperplane* function is obtained for $\alpha = 0$, given as $F_7(\mathbf{x})$ in (3.14). For $N = 2$, the contour plot of this function consists of isofitness lines parallel to each other. The general case of the hyperplane can be given as

$$F_{hp}(\mathbf{x}) \equiv F_{11}(\mathbf{x}) := c \cdot \mathbf{v}^{\mathrm{T}}\mathbf{x} - d, \qquad c \in \mathbb{R},\, \mathbf{v} \in \mathbb{R}^N \ , \tag{3.21}$$

where $\mathbf{v}$ indicates the unit vector in the direction of the gradient, therefore the direction of the largest fitness increase. The constant $-d$ determines the fitness value at the origin ($\mathbf{x} = \mathbf{0}$). For the case $c = 1$, $\mathbf{v} = (1, 0, \ldots, 0)^{\mathrm{T}}$, (3.21) becomes (3.14). Conversely, (3.21) is obtained from (3.14) by rotating the coordinate axes, and setting $c = 1$.

Note that the counting bits function (`OneMax`), $F_{1M}(\mathbf{x}) : \mathbb{B}^N \to \{0, \ldots, N\}$, $N \in \mathbb{N}$,

$$F_{1M}(\mathbf{x}) \equiv \texttt{OneMax}(\mathbf{x}) := \sum_{i=1}^{N} x_i \tag{3.22}$$

is a special case of (3.21), with $\mathbb{B}^N$ as the search space, $c = \sqrt{N}$, $d = 0$, and $\mathbf{v} = \frac{1}{\sqrt{N}}(1, 1, \ldots, 1)^{\mathrm{T}}$. The difference in the indexing of the variables x_i does not change the nature of the fitness function. `OneMax` is a well known function in the research area of genetic algorithms (GA). It is commonly used in the theoretical analysis of the convergence properties of GA.

The sharp ridge. The second important member of the ridge family is the *sharp ridge.* It is obtained for $\alpha = 1$, given as $F_8(\mathbf{x})$ in (3.15). In Figure 3.4, the sharp ridge is shown for $d = 0.01$ (left) and $d = 1$ (right), respectively ($N = 2$). The isofitness lines can be used for the visualization of the success regions, i.e. they indicate the regions with higher fitness values.

Obviously, for small values of d, the sharp ridge becomes similar to the hyperplane. For $0 < \alpha < 1$ and $\alpha < 0$, one obtains other ridge functions. These functions do not yield interesting results as far as the convergence measures (Chapter 4) are concerned, see Chapter 6 for more information.

The parabolic ridge. The third important member of the family is the *parabolic ridge.* It is obtained for $\alpha = 2$, given as $F_9(\mathbf{x})$ in (3.16). A gentle picture of it is shown in

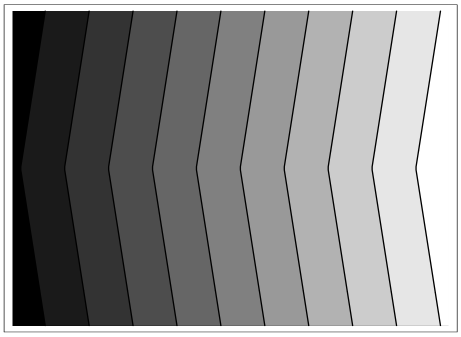 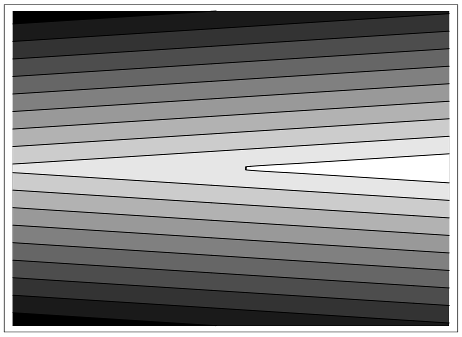

Figure 3.4: Two contour plots of the sharp ridge, for $d = 0.01$ (left) and $d = 1$ (right), with two variables ($N = 2$). The variable x_0 is indicated on the horizontal axis, x_1 on the vertical one. The brighter areas have higher fitness values. $F_8(\mathbf{x}) = x_0 - d\sqrt{x_1^2}$ is plotted here in the interval $\{x_0 \in \{0, 15\}, x_1 \in \{-85, 85\}\}$.

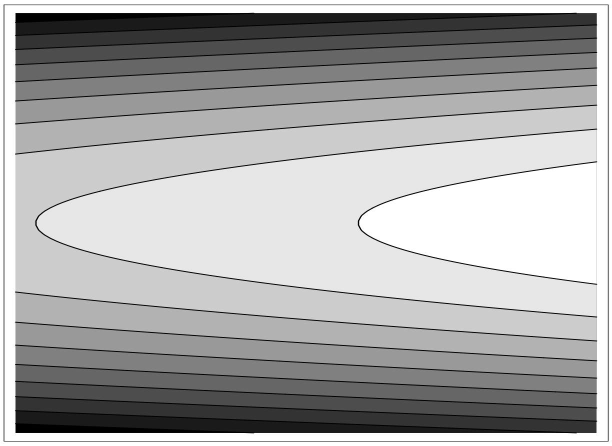

Figure 3.5: The contour plot of the parabolic ridge with two variables ($N = 2$), for $d = 0.01$. $F_9(\mathbf{x}) = x_0 - dx_1^2$ is plotted here in the interval $\{x_0 \in \{0, 15\}, x_1 \in \{-85, 85\}\}$. See also the caption of Figure 3.4.

Figure 3.5, for $N = 2$ and $d = 0.01$, using isofitness curves on a contour plot. A change in d affects the contour plot of the parabolic ridge case, as well as for other cases with $\alpha \neq 0$. Suppose that we are comparing the fitness values of two states, $\mathbf{x}_1$ and $\mathbf{x}_2$ in the search space, with different corresponding r values. The difference $F_R(\mathbf{x}_2) - F_R(\mathbf{x}_1)$ increases if d is increased. Therefore, any change in d should generate a different convergence behavior (see Chapter 4 for the definitions). However, for constant α, one can re-scale the axes so that the same contour plot is obtained. A similar rescaling is not possible for the case with constant d and variable α. As a result, one can foretell that α and d should influence the convergence behavior in different ways.

Based on the contour plots for $\alpha = 1$ and $\alpha = 2$, it becomes visually clear that the optimum is far right (maximize x_0), and on the x_0 axis (i.e. $\forall i \neq 0 : \hat{x}_i = 0$), as indicated in (3.18). Therefore, the x_0 axis is also named as the *progress axis* or the *ridge axis*. The variable r is called the *distance* to the progress axis.

The general rotated ridge function. Actually, the ridge axis does *not* have to coincide with the x_0 axis. An example is given in Figure 3.6. The ridge axis can be in any direction

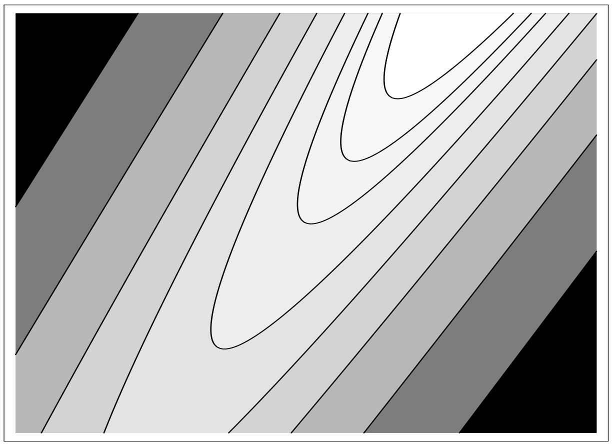

Figure 3.6: The contour plot of the general rotated parabolic ridge, with $d = 1$, $N = 2$, $\mathbf{v} = \frac{\sqrt{5}}{5}(1,2)^{\mathrm{T}}$. In this case, $F_{10}(\mathbf{x}) = \frac{\sqrt{5}}{5}(x_0 + 2x_1) - \left[\frac{\sqrt{5}}{5}\sqrt{(2x_0 - x_1)^2}\right]^2$ is plotted in the interval $\{x_0 \in \{-15, 15\}, x_1 \in \{-15, 15\}\}$. The isofitness lines are drawn for $F_{10}(x_0, x_1) \in \{-200, -100, -50, -25, -10, 0, 5, 10\}$. See also the caption of Figure 3.4.

of the N-dimensional search space. If one formalizes this direction using the unit vector $\mathbf{v}$, one obtains the general rotated ridge function $F_{RR}(\mathbf{x})$ as shown in Equation (3.17). The special case $\alpha = 0$ is already defined in (3.21). As it is done in (3.12) for $F_R(\mathbf{x})$, we can also define the distance to the ridge axis for $F_{RR}(\mathbf{x})$: the scalar quantity $\mathbf{v}^{\mathrm{T}}\mathbf{x}$ is analogous to the linear component x_0, and $\|(\mathbf{v}^{\mathrm{T}}\mathbf{x})\mathbf{v} - \mathbf{x}\|$ to r, respectively. Thereby one obtains an important result: The maximization of $\mathbf{v}^{\mathrm{T}}\mathbf{x}$ and the minimization of $\|(\mathbf{v}^{\mathrm{T}}\mathbf{x})\mathbf{v} - \mathbf{x}\|$ are *contradicting goals*. Differing from the aligned simple case given in (3.11), a change in a single variable influences both of these parts of the fitness function $F_{RR}(\mathbf{x})$. This difference between $F_{RR}(\mathbf{x})$ and $F_R(\mathbf{x})$ is caused only by the rotation of the search space itself. The contradicting goals for $F_{RR}(\mathbf{x})$ were denoted as long term and short term goals, respectively. The nature of these goals do not change by the rotation, since the shape of the fitness function remains unchanged. By using a simple rotation, one obtains $F_R(\mathbf{x})$ from $F_{RR}(\mathbf{x})$. An obvious side benefit of such a rotation is the *decoupling* of the subgoals, which may influence the performance of some optimization algorithms on the ridge functions. Both $F_R(\mathbf{x})$ and $F_{RR}(\mathbf{x})$ are considered in this work, see the chapters for theoretical and empirical results (Chapter 6 and Chapter 7).

Relation to the sphere model. Many theoretical results are obtained by the analysis of ES for the sphere model. If the relation of ridge functions to the sphere model is well-established, some theoretical results obtained for the sphere model can be used in the theoretical analysis of ridge functions. Comparing the definitions of r in (3.12) and of D in (3.6), respectively, one realizes that the variable r is nothing but D in $N-1$ dimensions (for $\hat{x}_i = 0$). In this $(N-1)$-dimensional subspace, we have a special case of the sphere model (see Figure 3.1). Additionally, r has generally much more influence than x_0 on the fitness function. The minimization of r has been named as the short term goal in this subsection. The earlier results on the sphere model are used in the chapter for the theoretical analysis (Chapter 6, Section 6.4).

Ridge functions for large α. Another important property of the ridge functions is observed on the isofitness lines as α goes to infinity (see Figure 3.7). For large α ($\alpha > 2$), the fitness function is dominated by the effect of r if $r > 1$. In this case, a movement in a direction perpendicular to the ridge axis causes a much greater change in the fitness value than a movement along. As a result, for $\alpha > 0$, the isofitness curves bend further toward the ridge axis as α is increased further. Consequently, the same quantitative increase in r has a larger effect on the fitness function for larger values of α: The picture of the function becomes like an *inclined razor*, that is most clearly observed on the contour plot for r-versus-x_0.

Eventually, the mathematicians define the "ridge function" as [Pin97]

$$F(\mathbf{x}) := f\left(\sum_{i=1}^{N} a_i x_i\right), \tag{3.23}$$

for some fixed choice of constants $a_i \in \mathbb{R}$, and using a one-variable function f. Therefore, f has constant values on the isofitness hypersurfaces of the respective hyperplane. The function family given in (3.23) is not considered in this work.

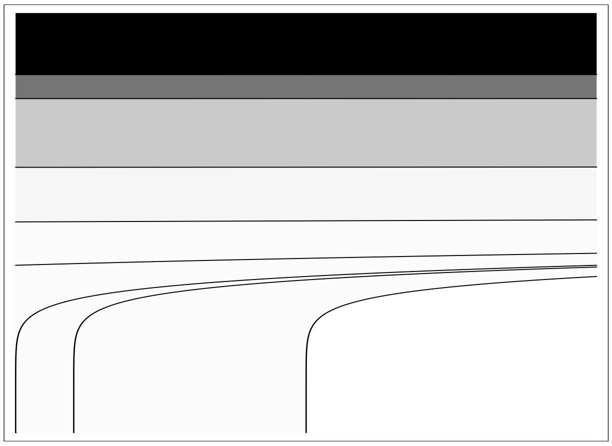

Figure 3.7: The contour plot of $F_R(\mathbf{x}) = x_0 - r^{10}$ ($\alpha = 10$, $d = 1$). The isofitness lines are drawn for $F_R \in \{-2 \cdot 10^6, -10^6, -10^5, -10^4, -10^3, 0, 100, 500\}$. The plot shows the interval $x_0 \in \{0, 1000\}$ on the horizontal axis and $r \in \{0, 5\}$ on the vertical axis. For $r < 1$, $F_R(\mathbf{x})$ is a function similar to the hyperplane $F_7(\mathbf{x})$; otherwise, the isofitness lines are almost parallel to the horizontal axis x_0.

3.3.3 The corridor models

The corridor model constitutes of a simple monotonically increasing function that is symmetrical around its corridor axis, and a set of constraints applied to it. Originally, the

$$
\begin{aligned}
F_C(\mathbf{x}) \equiv F_{11}(\mathbf{x}) &:= \begin{cases} cx_0 & \text{if all constraints fulfilled} \\ -\infty & \text{otherwise} \end{cases} && (3.24) \\
G_j(\mathbf{x}) &: \quad |x_j| \leq b, \qquad j = 1, \ldots, N-1 && (3.25) \\
H(\mathbf{x}) &: \quad \sqrt{\sum_{i=1}^{N-1} x_i^2} \leq b' && (3.26) \\
F_{12}(\mathbf{x}) &:= \begin{cases} c \cdot \mathbf{v}^{\mathrm{T}}\mathbf{x} & \text{if all constraints fulfilled} \\ -\infty & \text{otherwise} \end{cases} && (3.27) \\
H_2(\mathbf{x}) &:= \|(\mathbf{v}^{\mathrm{T}}\mathbf{x})\mathbf{v} - \mathbf{x}\| \leq b' && (3.28)
\end{aligned}
$$

Figure 3.8: The two corridor models are given with their constraints. $F_C(\mathbf{x})$ with the constraints $G_j(x)$ is named as *rectangular corridor*, with $H(x)$ the *cylindrical corridor*. The *rotated* cylindrical corridor model with the corridor axis $\mathbf{v}$ is given in (3.27), and its constraint $H_2(\mathbf{x})$ in (3.28). $c, b, b' \in \mathbb{R}^+$, $\mathbf{v} \in \mathbb{R}^N$, $\|\mathbf{v}\| = 1$.

hyperplane function is chosen as the monotonically increasing function [Rec71, Rec73]. The same case is considered in this work, as indicated in (3.24) and (3.27). The states of the search space that do not fulfill all corresponding constraints have a fitness value $-\infty$ (see also Subsection 2.6.3). The two corridor models of interest are introduced in Figure 3.8; they are explained in the following.

The corridor models are named after the shape of the feasible region imposed by the corresponding constraints. Only *rectangular* and *cylindrical* ones will be mentioned here. They have (3.25) and (3.26) as the constraints, respectively. $\mathbb{R}^+$ represents the set of positive real numbers, i.e. $b \in \mathbb{R}^+$ means "$b \in \mathbb{R} \wedge b > 0$".

The rectangular corridor model is named after the shape of the feasible region imposed by $N-1$ constraints given in (3.25). The width of each edge is $2b$; however, one can define a different corridor width b_i for each variable. The cylindrical corridor has a single constraint; $H(\mathbf{x})$ imposes a hyper-cylinder since x_0 is not bounded.

Actually, the corridor should not always be aligned with the x_0 axis. The rotated cylindrical corridor model $F_{12}(\mathbf{x})$ is given in (3.27), with its constraint $H_2(\mathbf{x})$ in (3.28).

The rectangular corridor model was introduced in [Rec73, p. 105]. Its definition can also be found in [Sch95, p. 134, p. 351] along with the rotated case. The cylindrical corridor is included in the large problem catalog of Schwefel [Sch75], [Sch95, p. 361].

Using the convergence measures to be introduced in Chapter 4, this work will investigate whether the corridor models can be considered as the limit cases of the ridge functions for $\alpha \to \infty$.

3.3.4 The demonstrative polynomial

This subsection will introduce a simple, one-dimensional polynomial. This fourth-degree polynomial

$$F_{13}(x) \;:=\; -3x^4 + 16x^3 + 66x^2 - 360x, \quad x \in \mathbb{R} \tag{3.29}$$

will serve in visualizing the convergence measures in Chapter 4. This fitness function will not be further investigated in other parts of this work. By differentiating $F_{13}(x)$ with respect to x,

$$\frac{\mathrm{d}F_{13}(x)}{\mathrm{d}x} = -12x^3 + 48x^2 + 132x - 360 = -12(x-2)(x-5)(x+3) \;, \tag{3.30}$$

one finds that $F_{13}(x)$ has its local optimum (maximum) for $x = 5$, and its (global) optimum for $x = -3$ ($\hat{x} = -3$, $\hat{F}_{13} = 999$). $F_{13}(x)$ can be extended for $N > 1$ as

$$F_{14}(\mathbf{x}) \;:=\; \sum_{i=1}^{N}(-3x_i^4 + 16x_i^3 + 66x_i^2 - 360x_i) \;, \tag{3.31}$$

with ($\hat{\mathbf{x}} = (-3, -3, \ldots, -3)^{\mathrm{T}}$, $\hat{F}_{14} = 999N$), being the global optimum. Note that this function $F_{14}(\mathbf{x})$ has 2^N local optima, of which one is global.

Chapter 4

Convergence measures

This short chapter introduces the convergence measures that are considered in this work. These measures consist of *progress measures*, *success measures*, and the distance r to the progress axis. The progress measures are the quality gain $\overline{Q}$, the progress rate φ, and the self-adaptation response ψ (SAR). The success is measured by using the success probability, denoted by P_{s1} for a single descendant and by $P_{s\lambda}$ for whole offspring population. The symbol r stands for the distance to the ridge axis, as introduced already in Subsection 3.3.2. It is repeated here for reasons of completeness.

The convergence measures can alternatively be classified according to the space where they are defined. Therefore, the progress rate φ and the distance r will be called the search space measures. Similarly, the quality gain $\overline{Q}$ and the success probabilities P_{s1} and $P_{s\lambda}$ will be called the fitness space measures. The self-adaptation response ψ cannot be classified in this way.

Before starting with the definitions of these measures, the approach chosen for the convergence analysis should be justified. As will be seen, these measures do *not* directly give the *convergence order*, i.e. the order of generations necessary to reach the optimum for a given ES algorithm. *Nor* do they directly ensure the *global convergence*, i.e. that the optimum will be reached independently of the initial state $\mathbf{P}^{(0)}$. The initial population $\mathbf{P}^{(0)}$ is defined in line L3 of the ES algorithm used, as described in Chapter 2. However, they describe the quantitative nature of the evolution of the population over generations. Convergence measures serve as a basis of the global convergence measures, e.g. the number of generations required to reach a pre-specified fitness value. Instead of global convergence measures, microscopic aspects of evolution are considered in this work.

These measures serve for the understanding how the ES algorithm functions; they explain the evolution over time on a microscopic scale, i.e. from one generation g to the next generation $g+1$. They explain the local behavior of the ES algorithm (local in time, *not* in space). Moreover, they contain enough information for answering the above-mentioned questions on the convergence order and global convergence (also called convergence reliability).

Actually, one can argue that the global behavior of an algorithm cannot be understood if the local behavior remains uncovered. Since the microscopic behavior *causes* the global behavior, the accuracy of the estimates on the global behavior are not trustworthy if they are derived without noticing the local behavior. Consequently, the influences of the evolutionary operators (selection, recombination, mutation, etc.) should essentially be investigated locally. For the investigation of these operators, only the local analysis based on statistical measures is of practical importance. Otherwise, global investigations ignoring the statistical quantities may lead to inaccurate, or practically unusable results. In the worst case, the interpretations of such results will trigger incorrect conclusions.

Another theoretical tool for describing the transition $\mathbf{P}^{(g)} \longrightarrow \mathbf{P}^{(g+1)}$ is the Markov process of first order. Since $\mathbf{P}^{(g+1)}$ only depends on $\mathbf{P}^{(g)}$, such a transition can be perfectly formalized using a Markov chain. This stochastic evolution can be expressed by Chapman-Kolmogorov equations [Fis76]. The population density of $\mathbf{P}^{(g+1)}$ can be stated using the parental population $\mathbf{P}^{(g)}$, the offspring population $\tilde{\mathbf{P}}^{(g)}$, and the parameters of the algorithm [Bey96c, p. 26]. However, such integral equations expressing $\mathbf{P}^{(g+1)}$ do not have a general analytical solution for any fitness function. Moreover, such population transitions do not describe the functioning mechanism of the ES algorithms, either. Therefore, although remaining theoretically important, they are of no practical merit.

Before starting with the definitions, two short remarks are necessary on the notation. In this work, $\mathbf{x}$ or $\mathbf{y}$ stand equivalently for the states in the search space. The vector $\mathbf{z}$ stands for the N-dimensional mutation vector, defined again in the search space.

4.1 Progress measures

Three progress measures will be introduced here. They are measured in three different spaces. The *quality gain* $\overline{Q}$ is measured in the fitness space, based on the scalar fitness function values. The *progress rate* φ is measured in the search space, based on the object variables of successive generations and of the optimum. The third measure is the *self-adaptation response* ψ. It is defined in the space of strategy parameters. The former two progress measures will be considered in this work. The third one is given for completeness.

These three measures correspond to the three different features of an individual, as described in Algorithm 5: $F(\mathbf{y}_m^{(g)}), \mathbf{y}_m^{(g)}$, and $s_m^{(g)}$. The ordering of the progress measures reflects the level of difficulty of the analysis: One may easily get very accurate formulae for $\overline{Q}$, as to be shown in the section dedicated to $\overline{Q}$ in Subsection 5.3.6, or in Section 6.1. However, the same accuracy for φ is bound with tedious and difficult analytical derivations. The problem is yet more severe for the third measure: Only empirical results of preliminary nature are obtained for ψ in the scope of this work.

4.1.1 Quality gain $\overline{Q}$

The quality gain $\overline{Q}$ is defined in the fitness space as

$$\overline{Q} := \mathrm{E}\left\{F^{(g+1)} - F^{(g)}\right\} = \mathrm{E}\{\Delta F\} \tag{4.1}$$

for $\mu = 1$. For the general case $\mu > 1$, $\langle F \rangle$

$$\langle F \rangle := \frac{1}{\mu} \sum_{m=1}^{\mu} F_m \tag{4.2}$$

should be used instead of F, and $\overline{Q}$ is defined using $\langle F^{(g+1)} \rangle$ and $\langle F^{(g)} \rangle$ then. This notation for the average value was already introduced in (2.12). Please note that knowing the optimum $\hat{F}$ is not necessary for the definition of $\overline{Q}$. Formally, $\overline{Q}$ is defined as the expected progress in the fitness space over a single generation. The case of maximization is considered in this work without loss of generality.

$\overline{Q}$ for the $(1 \overset{+}{,} \lambda)$-ES. For $\mu = 1$, $\overline{Q}$ can easily be specified further. For this purpose, we define another important measure in the fitness space. The *local quality function* $Q(\mathbf{z})$

$$Q(\mathbf{z}) \equiv Q_{\mathbf{x}}(\mathbf{z}) := \Delta F = F(\mathbf{x} + \mathbf{z}) - F(\mathbf{x}) \tag{4.3}$$

gives the difference of the fitness values between the parent and one of its descendants, generated by the mutation $\mathbf{z}$. The subscript $\mathbf{x}$ in the definition of this quantity should denote that this function is strongly dependent on the local conditions. In order to simplify the notation, $Q(\mathbf{z})$ will be used for the local quality function instead of $Q_{\mathbf{x}}(\mathbf{z})$.

Actually, $Q(\mathbf{z})$ is generated by a specific random vector $\mathbf{z}$. In the analysis, $Q(\mathbf{z})$ will be used to define the random variable Q induced by the mutation distribution. The derivation of an integral expression for $\overline{Q}$ will finish this subsection.

For $\mu = 1$, the quality gain $\overline{Q}$ is the expected value of the change in the fitness value of the parent individual in a single generation. Consequently, $\overline{Q}$ is the expected value for the difference of the fitness values of the parent individual and of its best descendant. Since we are trying to find the maximum of $F(\mathbf{x})$, this difference reduces to an expression based on $Q(\mathbf{z})$

$$\overline{Q} := \mathrm{E}\{F^{(g+1)} - F^{(g)}\} = \mathrm{E}\{F_{1;\lambda}^{(g)} - F^{(g)}\} = \mathrm{E}\{Q(\mathbf{z}_{1;\lambda})\} \ , \tag{4.4}$$

where $\mathbf{z}_{1;\lambda}$ stands for the mutation vector which generated the best descendant and $F_{1;\lambda}^{(g)}$ for its fitness value. This notation differs from the one used in Equation (2.9). It is introduced to denote the best sample of both minimization or maximization cases with the same symbol. For $\mu > 1$, it provides convenience in the notation of induced order statistics (see Page 67 and Page 105). The expected value of the local quality function for the *fittest* descendant can be stated as an integral

$$\overline{Q} := \mathrm{E}\{Q(\mathbf{z}_{1;\lambda})\} = \mathrm{E}\{Q_{1;\lambda}\} = \int_{Q' = Q_l}^{Q' = \hat{Q}} Q' p_{\lambda:\lambda}(Q') \mathrm{d}Q' \ . \tag{4.5}$$

In this equation, Q' denotes the random variable for the local quality function of the best offspring, and $p_{\lambda:\lambda}(Q')$ its probability density function. The upper limit of the integral is the maximum possible value for Q', $\hat{Q} = \hat{F} - F^{(g)}$. The lower limit Q_l depends on the selection strategy used: For the *plus* strategy, $Q_l = 0$ since the parent survives to the next generation if $F_{1;\lambda}^{(g)} < F^{(g)}$; therefore, $\overline{Q} \geq 0$ in this case. For the $(1, \lambda)$-ES, $Q_l = -\infty$ or $Q_l = \check{F} - F^{(g)}$, if $\check{F}$ should denote the worst attainable fitness value.

4.1.2 Progress rate φ

The *progress rate* φ defines the expected progress in the search space in one generation. It will be used to formalize the long term goal introduced in Subsection 3.3.2. More formally, it is the expected value of the decrease in the distance to the optimum in a single generation,

$$\varphi := \mathrm{E}\{\|\hat{\mathbf{x}} - \mathbf{x}^{(g)}\| - \|\hat{\mathbf{x}} - \mathbf{x}^{(g+1)}\|\} \ . \tag{4.6}$$

In this equation, $\hat{\mathbf{x}}$ stands for the object variables of the optimum. For the general case $\mu > 1$, $\mathbf{x}$ should be replaced by $\langle \mathbf{x} \rangle$, giving $\langle \mathbf{x} \rangle^{(g)}$ and $\langle \mathbf{x} \rangle^{(g+1)}$. It is important to note that φ is *not* the expected distance traveled in the search space, such as $\mathrm{E}\{\|\mathbf{x}^{(g)} - \mathbf{x}^{(g+1)}\|\}$.

For the ridge functions, the φ definition (4.6) reduces to the expected distance traveled in the direction of the progress axis (see Equation (3.18)). This follows immediately as a by-product of the discussion of the optimum of ridge functions which follows after Equation (3.18). It has been shown there that the optimum of ridge functions can be considered as a limit value and the ridge function as the limit of another function family. The same argumentation can be used here to simplify the progress rate formula (4.6) for ridge functions. For the other function family (3.19), the distance r to the ridge axis must be considered in the progress rate calculation. As a gets smaller, however, the change in r decreases in importance. If the optimum is at a very large distance in x_0 direction, a change in r direction does not reflect itself in φ as high as a change in x_0 direction. This can be proven by a respective Taylor expansion, however, it is omitted here. Therefore, for (3.19), the effect of r in the calculation of φ decreases as a gets smaller. Finally, for the limit $a \to 0$, it can be neglected. For this limit, one obtains the general ridge function from (3.19). As a result, if x_0 denotes the progress axis, one gets

$$\varphi = \mathrm{E}\{\Delta x_0^{(g)}\} = \mathrm{E}\{x_0^{(g+1)} - x_0^{(g)}\} \ . \tag{4.7}$$

Alternatively, without considering a limit process, this equation can also be seen as a redefinition of the progress rate. That is, as has been argued in [Bey96c, p. 28], in general the progress need not to be defined with respect to the optimum. It can also be defined in an arbitrary direction. For example, in case of the hyperplane test function (3.14), the progress is defined in the gradient direction.

Additionally, one should use $\langle x_0 \rangle$ instead of x_0 for $\mu > 1$. The ES algorithm selects the best μ individuals based on the fitness values as $\mathbf{P}^{(g+1)}$. However, the fittest individual does *not* necessarily have the largest contribution to φ. Therefore, $\overline{Q}$ is a direct result of the selection process, whereas φ is just one of its by-products. Another by-product of the selection process is the progress measure φ_R (cf. (6.20) and (6.177)). It will be introduced in Chapter 6 and not here, in order to avoid the confusion with the progress rate φ.

Demonstrative comparison of φ and $\overline{Q}$. The progress measures φ and $\overline{Q}$ will be demonstrated in Figure 4.1 using the univariable function $F_{13}(x)$ (Page 39). For $F_{13}(x)$,

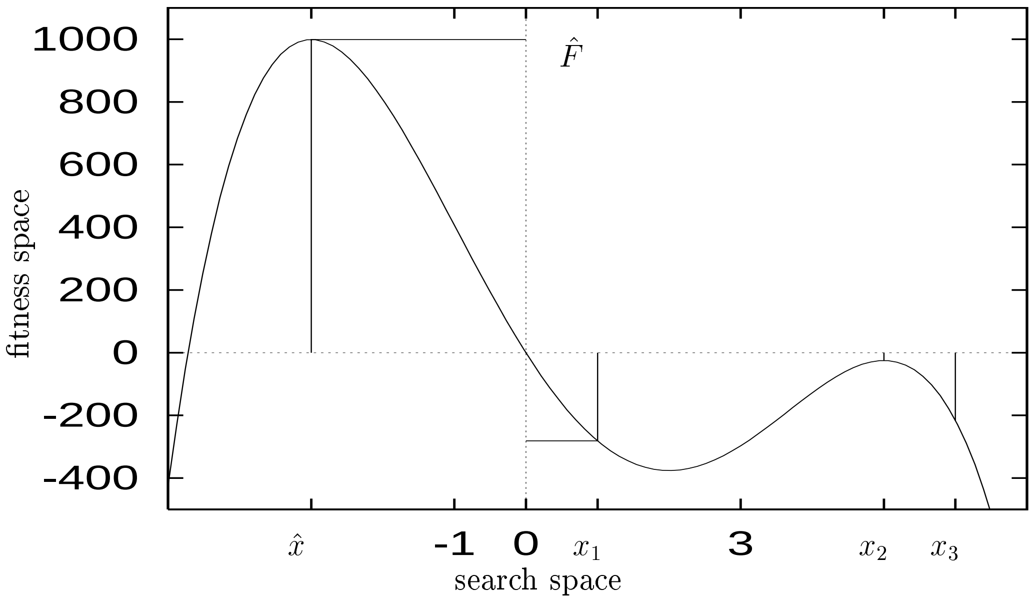

Figure 4.1: The comparison of the quality gain $\overline{Q}$ and the progress rate φ using the univariable fitness function $F_{13}(x)$. $F_{13}(x)$ was introduced in Equation 3.29 on Page 39. $(\hat{x}, \hat{F}_{13}) = (-3, 999)$. Some hypothetical movements on the search space will be used in considering these two progress measures. $x_1 = 1$, $x_2 = 5$, and $x_3 = 6$; with $F_{13}(x_1) = -281$, $F_{13}(x_2) = -25$, and $F_{13}(x_3) = -216$.

we have $\hat{x} = -3$, $\hat{F} = F_{13}(\hat{x}) = 999$; and a further local optimum at $x = 5$, $F(x) = -25$. In the following example, the three states $x_1 = 1$, $x_2 = 5$, and $x_3 = 6$ are considered in the search space. One can easily give the φ and $\overline{Q}$ values, *assuming* the movement denoted as $x^{(g)} \longrightarrow x^{(g+1)}$. The values for Δx can be read on the horizontal axis (search space), and for ΔF on the vertical one (fitness space). Three scenarios are given below.

1.	$x_3 \longrightarrow x_2$:	$\Delta x > 0$,	$\Delta F > 0$.
2.	$x_1 \longrightarrow x_2$:	$\Delta x < 0$,	$\Delta F > 0$.
3.	$x_3 \longrightarrow x_1$:	$\Delta x > 0$,	$\Delta F < 0$.

It is important to note that both φ and $\overline{Q}$ require a corresponding probability density function for their definition. This density function must cover all possible mutations. Therefore, the desired theoretical quantities cannot be obtained by stating just a single expected movement. That is, φ or $\overline{Q}$ values can only be computed using the respective probability density, as it will be done in Chapter 6.

Definitely, Scenario (3) cannot happen under elitist selection. The latter two scenarios indicate that φ and $\overline{Q}$ may have different tendencies in multimodal landscapes. Presumably, one expects principally different behavior for $\overline{Q}$ and φ.

The tendencies of these two progress measures will be analyzed in this work on unimodal functions. However, even on unimodal functions, $\overline{Q}$ and φ may show different characteristics. This becomes more clear after observing the contour plots given for the sharp ridge and the parabolic ridge in Subsection 3.3.2 (Figure 3.4 and Figure 3.5, respectively). In both plots, a movement along the vertical axis causes drastic changes in the fitness space; however, it yields exactly *no* progress toward the optimum in the search space. More formally, a change in r does not have any effect on φ (See Equation 3.12 and Section 4.3 for r). These figures indicate that generally $\varphi \neq \overline{Q}$. The empirical part verifying the theoretical analysis for arbitrary mutation vectors will follow in Chapter 7, yielding more surprising results.

4.1.3 Self-adaptation response ψ

In the previous two subsections, the progress measures $\overline{Q}$ and φ were introduced. The former one measures the progress in the fitness space, the latter in the search space, respectively. The third progress measure is the self-adaptation response ψ (SAR). It is of interest if the self-adaptation operator (L7, L8) is used in the ES algorithm.

As a result of self-adaptation of the mutation strength (σ-SA), the mutation distribution applied changes over generations. Principally, ψ depends on the state of the population at generation g, namely on $\mathbf{P}^{(g)}$. The definition of $\mathbf{P}^{(g)}$ is given in line L3 of the $(\mu/\rho, \lambda)$-ES algorithm with σ-SA, on Page 11. Since the mutation strength is expected to vary, the σ values of μ individuals in $\mathbf{P}^{(g)}$ are denoted by $s_m^{(g)}$.

This work only considers isotropic mutations. As can be seen in the literature, even the theoretical analysis of this case, and even for the $(1, \lambda)$-ES, is very complicated [Bey96c, Ch. 7], [Bey96b]. The theoretical quantity ψ measures the *expected relative change* in the mutation strength in a single generation

$$\psi = \psi(\mathbf{P}^{(g)}) := \mathrm{E}\left\{ \frac{s^{(g+1)} - s^{(g)}}{s^{(g)}} \,\middle|\, \mathbf{P}^{(g)} \right\} \quad . \tag{4.8}$$

Obviously, ψ depends on the local state $r^{(g)}$. For ridge functions, $r^{(g)}$ stands for the distance to the ridge axis. For a given $r^{(g)}$, $x_0^{(g)}$, and $s^{(g)}$, ψ can be used to predict the expected value of the mutation strength $s^{(g+1)}$ for the next generation. For $\mu > 1$, the average values of the respective quantities over μ parents should be used instead, i.e. $\langle s\rangle^{(g)}$, $\langle r\rangle^{(g)}$, $\langle x_0\rangle^{(g)}$, etc.

The special case $\psi = 0$ is observed for the ES algorithms without σ-SA. For the algorithm with σ-SA, this would mean that no change is expected in the value of the mutation strength. We have $\psi < 0$ if the mutation strength is expected to decrease. More formally, the expected value of $s^{(g+1)}$ will be less than $s^{(g)}$. This means that the descendants generated with the mutation strengths less than $s^{(g)}$ are expected to have relatively better fitness values. Consequently, they will be selected. Of course, depending on $r^{(g)}$ and $s^{(g)}$, one may also have $\psi > 0$. This case can be explained analogously.

The SAR gives monotonically decreasing functions of $s^{(g)}$ on the sphere model, using the σ-SA rule given in (2.16) [Bey96c, p. 276], [Bey96b]. The slope of ψ depends on the actual value of the strategy parameter τ used. Further information on τ will follow in Section 5.1.1. For large $s^{(g)}$ and small $D^{(g)}$ (residual distance to the optimum), one observes $\psi < 0$ for the sphere model, where sufficiently small $s^{(g)}$ will give $\psi > 0$. The self-adaptation response ψ is given here for completeness. It is not used in this work, i.e. the mutation strength is kept constant in the analysis.

4.2 Success measures

In the ES terminology, each mutation $\mathbf{z}$ applied to the parent $\mathrm{P}^{(g)}$ at $\mathbf{x}^{(g)}$ is called *successful* if $F(\mathbf{x}^{(g)} + \mathbf{z}) \geq F(\mathbf{x}^{(g)})$. Equivalently, this condition can be stated using the local quality function introduced in Equation 4.3, yielding $Q(\mathbf{z}) \geq 0$.

The probability of obtaining a successful individual at a given state is described formally by the *success probability.* For the $(1 + 1)$-ES case, the success probability is defined in [Rec73, p. 94]. For $\lambda > 1$, this probability can be computed for a single descendant or for all of the λ offspring created in a single generation. Both of these measures reflect different aspects of the analysis, and will be denoted by P_{s1} for the single descendant case, and by $P_{s\lambda}$ for the whole offspring population, respectively. Naturally, $P_{s\lambda}$ denotes having at least one successful descendant. Therefore, one can write

$$\begin{aligned} P_{s1} &:= P(Q(\mathbf{z}) \geq 0) && (4.9) \\ P_{s\lambda} &:= 1 - [1 - P_{s1}]^{\lambda} \ . && (4.10) \end{aligned}$$

The isofitness curves on the contour plots connect the states of the search space that have the same fitness value (Chapter 3). Therefore, they can be used in the visualization of a successful mutation. For $\mu > 1$, the success probabilities will be computed with respect to $\langle F \rangle^{(g)}$ in this work, i.e. $Q(\mathbf{z}) = \tilde{F}_l - \langle F \rangle^{(g)}$. Alternatively, $F(\langle \mathbf{x} \rangle^{(g)})$ could be used instead.

The P_{s1} values can easily be stated for the ridge functions at $r = 0$. For $\alpha > 0$, we have $P_{s1} \leq \frac{1}{2}$. For $\alpha > 1$, P_{s1} decreases for increasing mutation strength. For the special case $\alpha = 0$, the hyperplane, one has $P_{s1} = \frac{1}{2}$ (for any σ and r). The cases with $\alpha < 0$ and $r = 0$ are peculiar since they yield $P_{s1} \geq \frac{1}{2}$, decreasing down to $P_{s1} \gtrapprox \frac{1}{2}$ as $\sigma \to \infty$. The corresponding $P_{s\lambda}$ values always depend on λ. For example, $P_{s1} = \frac{1}{5}$ for $\lambda = 10$ corresponds to $P_{s\lambda} = 1 - (1 - 0.2)^{10} \approx 0.893$. Definitely, the success probability values change with respect to r for all ridge functions with $\alpha \neq 0$.

A last remark should be made on notation. If one has to give explicitly the actual value of λ *and* the ES algorithm used, the $P_{s\lambda}$ notation becomes cumbersome. As a result, one needs a simplification. For example, the $P_{s\lambda}$ value for the $(1, \lambda)$-ES with $\lambda = 10000$ will be denoted as $P_{s\,1,10000}$, and not as $P_{s10000\,1,10000}$.

4.3 Other measures

For the analysis of the sphere model, the measures mentioned in the previous two sections are sufficient. However, an additional measure is necessary for the analysis of the ridge functions: The *distance* to the progress axis (Equation 3.12 on Page 32). It is represented by r, its value for the special case $\mu = 1$ at generation g by $r^{(g)}$, and for $\mu > 1$ by $\langle r \rangle^{(g)}$.

The value of r is considered in the static, dynamic, and stationary analysis. These three cases are explained in the following section, Section 4.4. The value of $r^{(g)}$ influences the values of the progress and success measures $[\varphi, \overline{Q}, \psi, P_{s1}, P_{s\lambda}]$. This influence cannot be canceled out by normalization (see Section 4.4).

The effect of r on other convergence measures will be analyzed in this work (static analysis). Thereafter, the stationary value $R^{(\infty)}$ will be computed theoretically. Additionally, the investigation of $r^{(g+1)}$ for a given $r^{(g)}$ is also concerned in this work (dynamic analysis). The theoretical analysis follows in Section 6.4.

4.4 Final remarks

This section will summarize some remarks on the notation. Some of these notations were already introduced. Additionally, three important terms describing the type of analysis applied are explained in detail.

Optimal, peak performance. The optimum value of an observed quantity will be indicated by the *hat* "^" symbol. On Page 29 of Chapter 3, $\hat{\mathbf{x}}$ was used to denote the object variables of the optimum, and $\hat{F}$ for the (global) optimum. Similarly, one can introduce the symbols $\hat{\varphi}$, $\hat{\sigma}$, $\hat{\overline{Q}}$, etc. for the optimal value of the quantity concerned. The case of maximization is considered in this work without loss of generality.

Normalization. If the fitness function is analyzed for specific values of its parameters, the results obtained will also depend on these specific values. Naturally, more general results are more valuable since they indicate the characteristics of the convergence behavior in general.

The generalization of the results obtained are achieved by normalizations. For example, the following equation

$$\varphi^* := d^{\frac{1}{\alpha-1}}(N-1)\varphi, \qquad \sigma^* := d^{\frac{1}{\alpha-1}}(N-1)\sigma \tag{4.11}$$

introduces the normalizations used for the ridge functions. The *star* "*" symbol is used to indicate the normalized items. Naturally, any normalization is expected to simplify the formulae, and will be specific to the fitness function of interest. Therefore, they are not arbitrary: They strongly depend on the analytical formulae. Equation 4.11 will be introduced in the chapter for theory, in Section 6.3.

Indicating algorithms. In this work, the ES algorithms considered throughout the analysis will be indicated by a subscript to the convergence measures of concern, where necessary. For instance, this is the case when the results for different algorithms are compared, and when the results for the convergence measures should be given compactly, such as for $\varphi_{1\overset{+}{,}10}$, $\overline{Q}_{1+\lambda}$, $R^{(\infty)}_{1,\lambda}$, $P_{s1\,1\overset{+}{,}\lambda}$, $P_{s\,1+10}$, $P_{s\lambda\,1,\lambda}$, and $\psi_{1,10}$. The subscripts for the algorithms will be avoided if possible, in order to simplify the notation. In this case, the algorithm analyzed should be obvious from the context.

In the following, the three important categories of the analysis done will be mentioned. As one can infer, *static* is the opposite of *dynamic*; and *stationary* is a special case of the dynamics. After this overview, the formal definitions follow.

Static. As mentioned in Section 4.3, the values of the progress and success measures depend on $r^{(g)}$. Therefore, it is advisable to carry out the theoretical analysis for different r values. In other words, the results of these measures will be parameterized by r. Correspondingly, the empirical analysis is also made statically, as described in Subsection 2.6.1. In this case, the search space values of $\mathbf{P}^{(0)}$ are dictated by $\mathbf{x}^{(0)}$ in L3, and L14 is removed from the algorithm. Each generation of such a simulation run is called a "one-generation experiment".

Dynamic. Although the static case of the empirical analysis gives very accurate results, it does not give any information on the dynamic nature of the ES algorithm. For example, the static analysis explains the $r^{(g)} \longrightarrow r^{(g+1)}$ transition. Moreover, such transitions can be used to compute $r^{(g)} \longrightarrow r^{(g+k)}$ or $r^{(0)} \longrightarrow r^{(\infty)}$ type of transitions. However, for the empirical verification of these k-generation transitions, one needs corresponding experiments. For instance, one can verify the theoretical estimations for such transitions by averaging multiple experiments with the same starting condition. Additionally, if a finite limit is expected for a convergence measure, the verification by the empirical analysis is advisable. Furthermore, the number of generations required to attain this limit can be theoretically estimated and empirically verified. As to the ridge functions, r is such a measure (See the chapter on theory, Section 6.4). See Section 4.3 for the consequences.

Stationary. A special case of the dynamic analysis is called stationary. For example, the stationary value for r is analyzed in this work. As mentioned in the description of the dynamic analysis, the value of r averaged over generations for $g \to \infty$ goes to a limit. Thereafter, the $r^{(g)}$ values fluctuate around a value denoted by $R^{(\infty)}$. As the number of measurements is increased, the standard deviation of the time average of r over generations goes toward zero, although the fluctuations themselves do not vanish. The calculation of $R^{(\infty)}$ will be an important part of the chapter on theory (Section 6.4). For $\mu > 1$, the notation $\langle R\rangle^{(\infty)}$ may be used. However, this notation is too cumbersome, and therefore will be avoided where possible.

The number of generations necessary to reach $R^{(\infty)}$ is called the *transition period*, or more frequently the *transient time*. This period must be passed before collecting statistically relevant data on $R^{(\infty)}$. The measurements taken after this period are called *stationary*. The stationary data collected for the convergence measures differ slightly from the static measurements at $R^{(\infty)}$, caused by the fluctuations in r.

Chapter 5

State of Research

This chapter summarizes the state of research in three parts. Section 5.1 presents the historical development of the ES algorithms. Some hypotheses in the ES literature are collected in Section 5.2, filtering only those relevant for this underlying work. Section 5.3 has seven parts; it lists formulae and methods adopted from the ES literature. This last section establishes the basis of this work.

5.1 History of the ES

The details on the first decade of the ES can be found in [Rec73]. The reader is referred to the Handbook of Evolutionary Computation [BFM97] for more elaborate information on ES. A huge collection of bibliography on evolutionary computation can be found in [Ala94], and in later surveys of Alander. Additionally, several institutes make their literature lists and/or publications available on the Internet.

Evolution Strategies were first explored in the 1960s at the Technical University of Berlin by Bienert, Rechenberg, and Schwefel. The name "Evolution Strategies" covers a large number of algorithms; most of them introduced in Chapter 2. This section will give an overview of the ES history.

The idea of ES originated during the attempts of solving experimental, discrete-valued problems of hydrodynamics, see e.g. [Rec65]. Initially, the mutation distribution was binomial. The board of Sir Francis Galton (1822-1911) was used for generating these mutations [Rec73, p. 26]. The diploma thesis of Schwefel [Sch65] indicated the danger of stagnation if the binomial distribution is used; suggesting continuous distributions instead. In 1965, the first experiments on a Zuse Z23 computer were done by Schwefel.

The first algorithm of the ES family was the $(1+1)$-ES, introduced in 1964 by Rechenberg. In 1968, the $(1+1)$-ES was used in order to optimize the shape of a two-phase flashing nozzle [KS70]. The number of nozzle segments was also considered as a variable in this work. This was the first known application of the gene deletion and gene duplication operators.

Although the first experiments were done in discrete search spaces, the first theoretical results were obtained for high-dimensional ($N \to \infty$) real valued search spaces. Rechenberg introduced two fitness landscapes called sphere model and (rectangular) corridor model [Rec71]. For the $(1+1)$-ES case, he was able to carry out the theoretical analysis, and obtained asymptotically ($N \to \infty$) correct φ and P_{s1} values for isotropic normally distributed mutations. Also the noise-perturbed case is considered in his PhD thesis. In his analysis, he calculated the success rate values at $\hat{\varphi}$, and obtained $P_{s1} \approx 0.270$ for the sphere model and $P_{s1} = \frac{1}{2e} \approx 0.184$ for the corridor model. Based on these results, he devised the 1/5-th success rule. The 1/5-th success rule is principally attached to the $(1+1)$-ES in order to control the mutation strength upwards and downwards (see Subsection 2.6.11).

Moreover, Rechenberg argued that the behavior of the ES on any fitness landscape can be obtained at the first order approximation by observing its behavior on these two models. In other words, the sphere model is expected to approximate the neighborhood or the vicinity of the optimum, as far as the progress behavior is concerned, and the mutation strength should be decreased to come even closer to the optimum. If one is far away from the optimum, the progress behavior should be similar to the one observed for the (rectangular) corridor model. In this latter case, the optimum mutation strength could be set once and for all.

In [Rec73, p. 83-88], one can find the experimental application of the $(10+1)$-ES and of the $(10/2_{\mathrm{D}}+1)$-ES, as special cases of the $(\mu+1)$-ES and $(\mu/2_{\mathrm{D}}+1)$-ES, respectively. The aim of the experiment was getting a specific wing spot pattern for a butterfly model. The fitness value for each spot setting was given by a fitness value based on the Hamming distance to the desired pattern. The search space was $\mathbb{B}^{81}$. The colors white and black were symbolized on the symmetric wings as 0 and 1 on the respective position of a representing bit-string. Starting at an initial setting of all zeros, the number of generations necessary to reach a desired pattern is measured. The initial setting had a Hamming distance of 40 to the desired pattern. This experiment indicated that recombination can increase the progress rate. Another interesting fact on this experiment is how the mutation and recombination operators were applied. Since the search space was binary, the mutations are applied by flipping the bits with a certain probability. The recombination operator with $\rho = 2$ generated a bit-string in $\mathbb{B}^{81}$. The first bit is obtained from the first parent. Between each bit, the *parent* supplying the bit that codes the respective variable of the descendant can be changed with probability 1/2. The similarity of this recombination scheme to the uniform cross-over in genetic algorithms (GA) is remarkable, as well as the similarity of the mutation scheme to the mutations applied in GA.

The PhD thesis of Schwefel [Sch75] extended the theoretical analysis of the ES to the $(1 \overset{+}{,} \lambda)$-ES on the two above-mentioned models and on the hyperplane function. His thesis is reprinted [Sch77]; and translated to English with an additional part on correlated mutations [Sch81]. Later, its revised version is printed with including sections on genetic algorithms, simulated annealing, and tabu search [Sch95]. The notation "$(1 \overset{+}{,} \lambda)$" was used for the first time in his PhD thesis, in order to indicate both $(1, \lambda)$ and $(1+\lambda)$ at the same time in an elegant manner [Rec78, p.104]. Schwefel was able to determine the optimal number of descendants $\hat{\lambda}$ for the hyperplane, sphere model, and (rectangular) corridor model [Sch95, p. 127,133,141]. Moreover, the progress rate φ^* for the corridor model is also derived [Sch95, p. 139]. For the σ-SA, he proposed the log-normal self-adaptation, and the value $\tau \propto 1/\sqrt{N}$ [Sch95, p. 144]. Rechenberg proposed a different multiplicative self-adaptation rule as shown in (2.18) [Rec78]. In [Rec73, p. 135], Rechenberg proposed a further self-adaptation rule with three cases (increase mutation strength, do not change it, and decrease it). He investigated experimentally the sphere model and the (rectangular) corridor model for $N = 10$, and used a strategy vector for applying mutation.

Beyond the introduction of the log-normal self-adaptation, the introduction of the $(\mu \overset{+}{,} \lambda)$-ES in [Sch75], [Sch95, p. 119] opened further horizons to evolution strategies. The notation "$(\mu \overset{+}{,} \lambda)$-ES" is used to mention the (μ, λ)-ES and the $(\mu + \lambda)$-ES at the same time. Both of these algorithms are introduced in Schwefel's PhD thesis, and their performance is measured on several fitness functions by experiments. The $(\mu \overset{+}{,} \lambda)$-ES imitates the simultaneous character of the evolution, having a population of parents *as well as* of descendants. In his work, the performance of the ES was compared to several traditional optimization methods, e.g. hill-climbing strategies. The comparison is done on a large number of problems. Additionally, three recombination operators were introduced in his work. He showed that recombination provides additional benefit over the $(\mu \overset{+}{,} \lambda)$-ES in the progress behavior using comparative experiments. The overall performance of the ES was better than the others, whereas the algorithms specially tailored for some problems could surpass the ES at these.

Selection strategies other than the plus and comma selection can also be used in the ES. Schwefel introduced a more general scheme with finite life span κ [SR95]. It comprises the plus strategy for $\kappa = \infty$, and the comma strategy for $\kappa = 1$. This scheme was supposed to be advantageous for collectively adapting the mutation strength. The contemporary state of the single-population ES is also described in this work.

Experiments with several populations are also being done in the ES research. The hierarchical ES (Section 2.7) imitates the parallel (non-identically structured) populations. They were introduced in [Rec78], and implemented e.g. in [Her92]. Additionally, the notation "$(\mu/\rho, \lambda)$" was introduced in [Rec78].

5.1.1 Some recommendations

This subsection will summarize some advises and theoretical results related to the σ-SA. Additionally, it will give a tentative list of problem classes where the application of the ES (or generally, EA) is expected to give successful results.

The strategy parameters τ and β for the σ-SA were introduced in Subsection 2.5.1. The optimal values of these parameters depend on the fitness function, on the problem dimension N, and on the ES algorithm used. The $\tau \propto 1/\sqrt{N}$ scaling rule was already proposed in [Sch75], [Sch95, p. 144]. For the sphere model, the optimal value of τ for the $(1, \lambda)$-ES is given as [Bey96b], [Bey96c, p. 295]

$$\tau \simeq \frac{c_{1,\lambda}}{\sqrt{N}} \; . \tag{5.1}$$

The symbol (or constant) $c_{1,\lambda}$ is the progress coefficient for the $(1,\lambda)$-ES, which will be introduced in (5.18) of Subsection 5.3.4. The notation "$\simeq$" indicates that the absolute as well as the relative error is asymptotically zero; in this case for $N \to \infty, \lambda \to \infty$. Nevertheless, because of the approximations used in the analytical derivation, this formula for τ should be considered as a "*rule*", and not as a law. In order to generalize this result to the (μ,λ)-ES or to the $(\mu/\mu_{\mathrm{I}},\lambda)$-ES, one may use the respective progress coefficients. This estimate would be speculative, but could be used as the first rule of thumb.

For the value of β, Rechenberg proposes $\beta \approx 0.3$ [Rec94, p. 48]. He advises to decrease β for $N > 100$. After long analytical derivations, Beyer found the following correspondence principle [Bey96c, p. 291], [Bey96b]:

$$\tau^2 = \beta^2(1-\beta) \ . \tag{5.2}$$

Therefore, the value of β can be specified at least for the $(1,\lambda)$-ES case.

The σ-SA can also be applied to other mutation distributions. For the normal distribution, the mutation operators for the cases other than the isotropic one were mentioned in Subsection 2.6.6. The reader is referred to [SR95] for the self-adaptation of the mutation vector or the mutation (correlation) matrix.

When to use ES? In other words, when should we *prefer* using evolution strategies? In [Rec71],[Rec73, p. 126], it is proposed that the ES attains larger progress rates than the gradient strategies for $N \gtrapprox 4$. In [Rec94, p. 218], four principal conditions are listed at which the application of the ES is advisable. These principles also hold to some extent for other evolutionary algorithms:

1. If the fitness landscape is fissured, extremely unsmooth, or noise-perturbed.

2. If the number of variables is large ($N > 20$).

3. If the problem is new and the optimum or optimum variables is unknown.

4. If no problem-specific optimization algorithm is known.

These principles clearly describe the advantages of the ES.

A characteristic property of the ES should be mentioned at the end. From the beginning, all variables are mutated in the ES algorithm for generating a descendant. This is a significant difference between the ES and algorithms that mutate a single variable on the average (such as the breeder genetic algorithm (BGA)), caused by their specific mutation operator. The performances of these two mutation philosophies differ significantly if the object variables are correlated. Such correlations cause epistatic effects, requiring the mutation of two or more variables simultaneously for any progress in the search space coupled with the fitness increase. The reader interested in an empirical comparison of the performances of the $(1,10)$-ES and the BGA is referred to [Sal96].

5.2 Hypotheses in the ES literature

This section summarizes several hypotheses in the ES literature. Only a few of the important works in the ES literature are considered here. The hypotheses mentioned below are all related to this work, and this criterion was considered in the selection. Further hypotheses can be found in the literature.

5.2.1 Evolution window

The definition of the evolution window (in German: Evolutionsfenster) can be found in [Rec71], reprinted as [Rec73]. In [Rec73, p. 139ff], or [Rec94, p. 37], the results obtained for the sphere model and corridor model are generalized to "universal laws". According to one of these laws, a positive progress rate φ^* is only obtainable for an interval of the mutation strength $\sigma^* \in [a, b]$, where $0 < a < b < \infty$. This interval is called evolution window. The value of the optimal mutation strength $\hat{\sigma}^*$ should be in this interval, whereas $a \approx \hat{\sigma}^*/k$ and $b \approx k\hat{\sigma}^*$ are supposed to hold practically [Rec73, p. 140], where $k \propto \sigma\sqrt{N}$. For $\sigma^* \lessapprox a$, the value of φ^* approaches zero very fast; and for $\sigma^* \gtrapprox b$, it may even become negative.

5.2.2 The diffusion along the gradient path

In [Rec94, p.75,130,etc.], it is claimed that the populations of the ES algorithms follow the path dictated by the local gradient in the search space. The deviations from this path are explained by the stochastic fluctuations. It will be an interesting task to investigate whether the ES algorithms follow this path on the case of ridge functions. In [Sch75, p. 257], this hypothesis is substantiated by the fact that the ES avoids a systematic sampling over the whole search space. It would be interesting to investigate whether the ES algorithm follows the gradient direction for very small values of the mutation strength, i.e. $\sigma \to 0$.

Rechenberg stated in [Rec73, p. 127] that the maximal possible gain is achieved by progressing in the direction of the steepest ascent, in other words, in the direction shown by the gradient. This assumption seems to be plausible on the search space, provided that the mutation strength is sufficiently small. However, if ridge functions are considered, one observes that this direction does not always coincide with the direction toward the optimum. Furthermore, the direction of the gradient is not necessarily directed toward the optimum in all fitness landscapes. Therefore, this principle should be formalized further.

5.2.3 Elitist strategies and the progress rate

The nature of elitist strategies is perfectly described in [Rud97, p. 185] by the example of the $(1 \overset{+}{,} \lambda)$-*EA* as follows:

> "*Since a* $(1 + \lambda)$-*EA rejects the best offspring if it is worse than its parents, the expected improvement of the* $(1 + \lambda)$-*EA is at least as large as the improve-*

ment of the $(1, \lambda)$*-EA, provided that the same step size rules and the mutation distributions are used.*"

As the only difference to the $(1 \overset{+}{,} \lambda)$-ES, the $(1 \overset{+}{,} \lambda)$-*EA* defined in [Rud97] uses spherically symmetric distribution instead of the normal distribution in generating mutations. The *Central Limit Theorem* [Roh76, p.282] can be used to establish the relationship of the *EA* to the evolution strategy. For $(N \to \infty)$, isotropic mutations are asymptotically distributed on a hyperspherical shell, where the variance of their length decreases for increasing N. Therefore, this quoted statement is also expected to be valid for the ES. In the fitness space, the correctness of this statement is obvious, if the quality gain $\overline{Q}$ is measured statically. For the stationary $\overline{Q}$ measurement, as well as for the value of the progress rate φ, this hypothesis must be revisited. From the context, one may also conclude that this statement on the $(1 \overset{+}{,} \lambda)$-*EA* is only intended for the progress measures in the fitness space, and on a specific fitness function type, and cannot be generalized to any other fitness function.

Two important remarks in [Sch75] should be repeated here for completeness. Schwefel argued that the strategies which do not allow a worsening in the fitness value in the subsequent generation (namely, the elitist strategies) may search in wrong directions [Sch75, p. 256]. Furthermore, he added that a worsening in the fitness values at the subsequent generation should be allowed if the search is stagnated, remarking the importance of a limited life span [Sch75, p. 264]. Such a worsening (i.e. $\overline{Q} < 0$) is not allowed by the plus strategy. These two hypotheses of Schwefel will be considered in the analysis of ridge functions.

5.2.4 The universal progress law

Rechenberg postulated in [Rec94, p. 60] the universal progress rate law of the evolution strategy:

$$\Phi = \Delta - \Delta^2 \tag{5.3}$$

In this formula, Φ stands for the normalized progress rate φ, and Δ for a normalized quantity proportional to the normalized mutation strength σ.

Equation 5.3 is expected to give negative values as the normalized mutation strength goes to infinity. Since this formula is derived using the information on the fitness space, it is expected to reflect the tendency of the quality gain, i.e. $\overline{Q} < 0$ for $\sigma \to \infty$. Additionally, the value of the local gradient vector is also used in the derivation of (5.3). According to the considerations in Section 4.1, the derivation of φ based on the values of $\overline{Q}$ is generally not possible. This work will try to enlighten whether Equation (5.3) is relevant to ridge functions.

5.2.5 Limit cases for the fitness landscapes

The convergence measures were introduced in Chapter 4. These are used to measure different aspects of the optimization process. The convergence behavior can be formalized using these measures, and this behavior depends among others also to the fitness function analyzed. If it would be possible to construct fitness landscapes having certain extreme values for these measures, the analysis of any fitness function could be reduced to these specific cases. In the ES theory, a well known aim was the construction of typical cases for the fitness landscapes as well as the convergence measures are concerned. In [Rec73, p. 104], these were also called "extreme cases". Especially the values of φ and P_{s1} on different fitness functions played an important role in these considerations. In the historical development, the sphere model was accepted to describe landscapes in the neighborhood of the optimum, and the (rectangular) corridor model landscapes distant from the optimum, respectively. These two fitness functions are expected to stand at the ends of a scale of possible properties; therefore, any given fitness function is approximated in the first order by these two models. In [Sch75], a counter-example to this hypothesis can be found; the $\hat{\lambda}$ value for the hyperplane is not between the ones obtained for the corridor model and the sphere model. The progress rate figures of the ridge functions will be used to investigate this hypothesis in detail.

5.2.6 Evolutionary Progress Principle (EPP)

This principle formulated by Beyer states that the evolutionary progress consists of a gain part and a loss part [Bey96c, p. 19], [Bey97]. This principle is explained in this subsection for the search space, although it is also valid for the fitness space. The gain part in the EPP indicates the progress toward optimum. The loss part is caused as an inevitable consequence of the movements perpendicular to this progress direction. The actual form of these parts depend on the fitness function used; however, the loss part always increases faster than the gain part with respect to the mutation strength.

By observing the progress rate formulae for different ES algorithms on the sphere model (Point 5.3.5.2), one can easily identify that the progress rate has a positive term that is linear in σ^*, which can easily be identified as the gain part. The other term in the asymptotic ($N \to \infty$) formulae is negative, and quadratic in σ^* (for the sphere model). A similar observation is also true for the quality gain formulae.

At the first sight, one may argue that EPP is equivalent to the universal progress law, mentioned in Subsection 5.2.4. However, the arithmetic structure of the gain and loss parts is not postulated in EPP. These two parts are also expected to emerge in the progress rate formulae of ridge functions. The exceptional cases $\alpha = 0$ and $\alpha < 0$ of ridge functions –which do not have a loss part– will be considered as well.

5.2.7 Genetic Repair Hypothesis (GR)

For most fitness landscapes, it is observed that the progress rate for the $(\mu/\rho_{\mathrm{I}}, \lambda)$-ES or for the $(\mu/\rho_{\mathrm{D}}, \lambda)$-ES is larger than the one of the (μ, λ)-ES. The introduction of the recombination operator mostly increases the progress rate. This can be explained by the decrease in the loss term, named as *genetic repair* [Bey95a], [Bey96c, p.235-241]. In [Bey96c, p. 218-219], it is stated that the recombination operator is *not* expected to increase the progress rate φ if $P_{s1} = 1/2$ or $P_{s1} > 1/2$. The former statement is true for the hyperplane function ($\alpha = 0$), and the latter one for $\alpha < 0$. In both of these landscapes, Beyer advises to increase the mutation strength σ to attain a larger φ. A more formal explanation to this hypothesis can be found in the progress rate formula of the the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES on the sphere model (Point 5.3.5.2) and in the ones on ridge functions (Chapter for theory, Subsection 6.3.3).

5.2.8 Mutation induced speciation by recombination (MISR)

A special feature of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES is observed if the selection operator is switched off; e.g. for the $\mu = \lambda$ case [Bey96c, p. 235-241], [Bey97]. In this case, although the population starts to walk randomly in the search space, it does not diffuse arbitrarily. The average standard deviation of the population, as well as the transient time to attain this standard deviation, can be approximated analytically. This phenomenon can also be observed on the selection-invariant variables of the individuals.

5.3 Background

This section consists of some formulae which are derived in earlier works of the ES theory. The derivation of these formulae will be avoided here, giving just the citations to the relevant literature. These formulae will be used primarily in the chapter for theory, Chapter 6. This section has seven parts. Some formulae related to the normal distribution are given in Subsection 5.3.1. The local quality function (LQF) is formally introduced in Subsection 5.3.2. A theoretical formula related to the success probability P_{s1} is stated in Subsection 5.3.3. In Subsection 5.3.4, the progress coefficients that inevitably occur in the progress rate and quality gain formulae are given. Subsection 5.3.5 summarizes the progress rate formulae of the fitness functions relevant to this work. A technique for obtaining the quality gain formulae is shortly described in Subsection 5.3.6. Lastly, a method based on *induced order statistics* is summarized for the reader in Subsection 5.3.7. This method will be used in calculating expected values in the search space.

5.3.1 The normal distribution

This subsection will repeat some definitions from probability theory. These definitions are given for completeness, they can be found in the relevant literature, e.g. [Fis76, BS91]. The definition of the normal distribution is followed by the values of its moments, some important integral equalities, and the definition of Hermite polynomials.

The normal distribution is used in the implementation of the mutation operator. The N-dimensional probability density function of the normal distribution was already introduced in (2.3). The one-dimensional case and the corresponding cumulative distribution function $\Phi(x)$ after the normalization $x := \frac{z}{\sigma}$ can be stated as follows

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} \mathrm{e}^{-\frac{1}{2}\left(\frac{z}{\sigma}\right)^2} , \tag{5.4}$$

$$\Phi(x) := P(t < x) = \frac{1}{\sqrt{2\pi}} \int_{t=-\infty}^{t=x} \mathrm{e}^{-\frac{1}{2}t^2} \mathrm{d}t \ . \tag{5.5}$$

The error function $\mathrm{erf}(x)$ is also used in the literature as an alternative to $\Phi(x)$. They can be converted to each other as follows

$$\mathrm{erf}(x) := \frac{2}{\sqrt{\pi}} \int_{t=0}^{t=x} \mathrm{e}^{-t^2} \, \mathrm{d}t \ , \tag{5.6}$$

$$\Phi(x) = \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right) , \qquad \mathrm{erf}(x) = 2\Phi(\sqrt{2}x) - 1 \ . \tag{5.7}$$

The normal distribution with zero mean is denoted by $\mathcal{N}(0, \sigma^2)$. The standard normal distribution $\mathcal{N}(0, 1)$ is obtained simply for $\sigma = 1$. Its k-th moment $\overline{x^k} := \mathrm{E}\{x^k\}$ is zero if k is odd. Otherwise, $\overline{x^k}$ for even k reads

$$\overline{x^k} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x^k \mathrm{e}^{-\frac{1}{2}x^2} \, \mathrm{d}x = 1 \cdot 3 \cdots (k-1) \ . \tag{5.8}$$

Similarly, the moments of $\overline{y^k}$ of $\mathcal{N}(0, \sigma^2)$ are $\overline{y^k} = \overline{x^k}\sigma^k$.

Integral equalities. The integral expressions containing the normal distribution must be solved for the analytical derivation of the progress rate φ. The following three equalities derived in [Bey96c, Appendix A1, p.322] are used in this work:

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}t^2} \mathrm{e}^{-\frac{1}{2}(at+b)^2} \, \mathrm{d}t = \frac{1}{\sqrt{1+a^2}} \exp\left(-\frac{1}{2}\frac{b^2}{1+a^2}\right) \tag{5.9}$$

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t\mathrm{e}^{-\frac{1}{2}t^2} \mathrm{e}^{-\frac{1}{2}(at+b)^2} \, \mathrm{d}t = -\frac{ab}{(1+a^2)^{\frac{3}{2}}} \exp\left(-\frac{1}{2}\frac{b^2}{1+a^2}\right) \tag{5.10}$$

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}t^2} \Phi(at+b) \, \mathrm{d}t = \Phi\left(\frac{b}{\sqrt{1+a^2}}\right) \tag{5.11}$$

Hermite polynomials. The quality gain formula (5.39) is derived using Hermite polynomials $\mathrm{He}_k(x)$ [Fel71, p. 532], [Bey94], [Bey96c, p. 329]

$$\mathrm{He}_k(x) := (-1)^k \, \mathrm{e}^{\frac{1}{2}x^2} \cdot \frac{\mathrm{d}^k}{\mathrm{d}x^k} \mathrm{e}^{-\frac{1}{2}x^2} , \tag{5.12}$$

$$\mathrm{He}_k(x) = x^k - 1 \cdot \binom{k}{2} x^{k-2} + 1 \cdot 3 \cdot \binom{k}{4} x^{k-4} - 1 \cdot 3 \cdot 5 \cdot \binom{k}{6} x^{k-6} + \ldots \tag{5.13}$$

They also occur in the more accurate success probability formula (Subsection 5.3.3).

5.3.2 The local quality function (LQF)

The local quality function $Q(\mathbf{z})$ was introduced in (4.3) in order to further formalize the quality gain $\overline{Q}$ in the $(1 \overset{+}{,} \lambda)$-ES case. The LQF itself will be formalized here.

The fitness value $F(\mathbf{x}+\mathbf{z})$ can be approximated according to Taylor by expanding the fitness function at $F(\mathbf{x})$. The values of first and second order derivatives can be expressed using the gradient vector $\mathbf{a} := \nabla F(\mathbf{x})$ and the matrix $\mathbf{Q}$, respectively. Using this scheme, $Q(\mathbf{z})$ can be neatly described [Bey94], [Bey96c, p. 35]:

$$F(\mathbf{x}+\mathbf{z}) = F(\mathbf{x}) + \sum_{i=1}^{N} \frac{\partial F}{\partial x_i} z_i + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{\partial^2 F}{\partial x_i \partial x_j} z_i z_j + \dots \tag{5.14}$$

$$F(\mathbf{x}+\mathbf{z}) - F(\mathbf{x}) = Q(\mathbf{z}) := \mathbf{a}^{\mathrm{T}}\mathbf{z} - \mathbf{z}^{\mathrm{T}}\mathbf{Q}\mathbf{z} \tag{5.15}$$

$$(\mathbf{a})_i := \frac{\partial F}{\partial x_i}, \qquad (\mathbf{Q})_{ij} := -\frac{1}{2} \frac{\partial^2 F}{\partial x_i \partial x_j} \tag{5.16}$$

This approximation is exact if all third and higher order derivatives of F vanish, e.g. for the parabolic ridge $F_9(\mathbf{x})$. The vector $\mathbf{a}$ and the matrix $\mathbf{Q}$ will also be used in the definition of the quality gain $\overline{Q}$.

5.3.3 The success probability P_{s1}

The success probability P_{s1} was defined in (4.9) as $P_{s1} := P(Q(\mathbf{z}) \geq 0)$. Actually, it can be expressed as $P_{s1} = 1 - P(Q(\mathbf{z}) < 0)$. The aim here is to approximate the local quality function $Q(\mathbf{z})$ by a probability density function (pdf), in the scope of the *Central Limit Theorem* [Roh76, p.282]. The random variable of this pdf is symbolized by Q. If one names the mean value and the standard deviation of Q as M_Q and S_Q, respectively, one may introduce a standardized variable $z := \frac{Q - M_Q}{S_Q}$ with zero mean and variance one, $P_{s1} := 1 - P_z(z < 0)$. However, even after the standardization, the higher order moments of Q will differ from the ones of $\mathcal{N}(0,1)$. The cumulants κ_k reflect this difference in the skew, kurtosis, and higher order moments. The Hermite polynomials (Subsection 5.3.1) are used for this adaptation. The cdf $P_z(z)$ can be expressed by using this polynomial series by [Bey94], [Bey96c, p.116]

$$\begin{aligned} P_z(z) = \Phi(z) - \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\frac{1}{2}z^2} \Bigg[& \frac{\kappa_3}{3!}\mathrm{He}_2(z) + \left(\frac{\kappa_4}{4!}\mathrm{He}_3(z) + \frac{\kappa_3^2}{2 \cdot 3! \cdot 3!}\mathrm{He}_5(z) \right) \\ & + \left(\frac{\kappa_5}{5!}\mathrm{He}_4(z) + \frac{\kappa_3 \kappa_4}{3! \cdot 4!}\mathrm{He}_6(z) + \frac{\kappa_3^3}{(3!)^4}\mathrm{He}_8(z) \right) + \dots \Bigg] . \end{aligned} \tag{5.17}$$

The symbol κ_k stands for the k-th semi-invariant or k-th *cumulant* of the standardized $Q(\mathbf{z})$ distribution. Naturally, they would be zero if the standardized $Q(\mathbf{z})$ distribution would be $\mathcal{N}(0,1)$. The formulae for κ_3 and κ_4 will be given in Subsection 5.3.6. They serve as the correction to the skew and kurtosis, respectively. The second line of (5.17) can often be neglected for practical purposes. As a first order estimate, $P_{s1} \approx \Phi(M_Q/S_Q)$ will be used.

5.3.4 The progress coefficients

The progress coefficients inevitably occur in the progress rate and quality gain formulae of the ES algorithms. They stand for integral expressions that usually cannot be solved analytically for any given λ (number of descendants). The definitions for the progress coefficients will be given here. The values of these coefficients can be obtained by numerical integration. Alternatively, the tabulated values of $c_{1,\lambda}$, $c_{\mu,\lambda}$, and $c_{\mu/\mu,\lambda}$ can be found e.g. in [Rec94, Bey95a, Bey95b, Bey96c]. The subscripts of these three coefficients indicate the corresponding ES algorithm; the third one is obtained for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES. The analytical formulae can be found in [Bey96c, p. 71], [Bey96c, p. 184], and [Bey96c, p. 241], respectively. The $d^{(k)}_{1,\lambda}$ coefficient [Bey96b], [Bey96c, p. 117] emerges in the quality gain analysis of the $(1, \lambda)$-ES, and the equality $d^{(1)}_{1,\lambda} = c_{1,\lambda}$ holds; for example $d^{(2)}_{1,10} \approx 2.7121$ and $d^{(3)}_{1,10} = 5.3158$. There is also a $d^{(k)}_{1+\lambda}$ function [Bey96c, p. 118], corresponding to the $(1 + \lambda)$-ES; however, it will not be presented here since the analysis of the quality gain $\overline{Q}$ for the $(1 + \lambda)$-ES will not be considered in this work. The $e^{\alpha,\beta}_{\mu,\lambda}$ coefficient [Bey95b], [Bey96c, p. 167] occurs in the analysis of the (μ, λ)-ES. For the special case $e^{1,0}_{\mu,\lambda}$, one obtains the progress coefficient $c_{\mu/\mu,\lambda}$ [Bey95a], [Bey96c, p. 211]. The approximate analytical formula of $c_{\mu,\lambda}$ is complicated: It contains *nine* different $e^{\alpha,\beta}_{\mu,\lambda}$ coefficients [Bey95b], [Bey96c, p. 184]; therefore, it will be omitted here. The $c_{\mu,\lambda}$ values obtained from the simulations will be used where necessary. The coefficients relevant to this work read

$$c_{1,\lambda} \;:=\; \frac{\lambda}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t\, \mathrm{e}^{-\frac{1}{2}t^2} \left[\Phi(t)\right]^{\lambda-1} \,\mathrm{d}t \;, \tag{5.18}$$

$$d^{(k)}_{1,\lambda} \;:=\; \frac{\lambda}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t^k\, \mathrm{e}^{-\frac{1}{2}t^2} \left[\Phi(t)\right]^{\lambda-1} \,\mathrm{d}t \;, \tag{5.19}$$

$$e^{\alpha,\beta}_{\mu,\lambda} \;:=\; \frac{\lambda-\mu}{\left(\sqrt{2\pi}\right)^{\alpha+1}} \binom{\lambda}{\mu} \int_{-\infty}^{\infty} t^{\beta}\, \mathrm{e}^{-\frac{\alpha+1}{2}t^2} \left[\Phi(t)\right]^{\lambda-\mu-1} \left[1-\Phi(t)\right]^{\mu-\alpha} \,\mathrm{d}t \;, \tag{5.20}$$

$$c_{\mu/\mu,\lambda} \;:=\; \frac{\lambda-\mu}{2\pi} \binom{\lambda}{\mu} \int_{-\infty}^{\infty} \mathrm{e}^{-t^2} \left[\Phi(t)\right]^{\lambda-\mu-1} \left[1-\Phi(t)\right]^{\mu-1} \,\mathrm{d}t \;. \tag{5.21}$$

Additional to these formulae, two small tables for $c_{1,\lambda}$, $c_{\mu,\lambda}$ and $c_{\mu/\mu,\lambda}$ will be given on Table 5.1 (Page 62). The few values in these tables may help the reader to get a feeling on the order of these coefficients. On the table right, the $c_{\mu/\mu,10}$ and $c_{\mu,10}$ values are listed for $1 \le \mu \le 10$. These values are obtained from simulation runs with $G = 102000$, where the first 2000 generations served as transient time. The standard error for the $c_{\mu/\mu,10}$ values is around 0.0004, for the $c_{\mu,10}$ values around 0.0005. The theoretical values agree for $c_{\mu/\mu,10}$, whereas the theoretical $c_{\mu,10}$ are about one to two per cent larger, caused by the analytical approximations. The following values should be mentioned for completeness: $c_{1/1,10} = c_{1,10}$ for $\mu = 1$, and $c_{10/10,10} = c_{10,10} = 0$ for $\mu = \lambda = 10$. Obviously, one observes $c_{\mu/\mu,10} \le c_{\mu,10}$, although this ordering relation is not formally proven yet; and the equality only holds for $\mu = 1$ and $\mu = \lambda$. Any c coefficient can be determined experimentally by measuring the progress rate φ of the respective ES algorithm on the hyperplane fitness function for $\sigma = 1$ and $N = 1$. The progress rate formulae of the hyperplane can be found in Point 5.3.5.1.

Table 5.1: A collection of progress coefficients: $c_{1,\lambda}$, $c_{\mu/\mu,10}$, and $c_{\mu,10}$. On the left table, the $c_{1,\lambda}$ values are listed for some selected λ, obtained from [Rec94, p. 236-240]. All other $c_{1,\lambda}$ values ($1 \leq \lambda \leq 1000$) can be found in the table cited. A smaller table for selected λ values between 1 and 10000 is given in [Bey96c, p. 351]. The values of $c_{\mu/\mu,10}$ and $c_{\mu,10}$ coefficients can be obtained from the right table. Please note that $c_{10/10,10} = c_{10,10} = 0$ and $c_{1/1,10} = c_{1,10}$.

λ	$c_{1,\lambda}$
1	0
2	0.564189583548
5	1.162964473641
10	1.538752730835
20	1.867475059798
50	2.249073629390
100	2.507593636442
200	2.746042447452
500	3.036699345857
1000	3.241435770486

μ	$c_{\mu/\mu,10}$	$c_{\mu,10}$
1	1.539	1.539
2	1.270	1.350
3	1.065	1.187
4	0.893	1.041
5	0.739	0.902
6	0.595	0.765
7	0.456	0.625
8	0.317	0.476
9	0.171	0.296
10	0	0

5.3.5 The progress rate formulae

The determination of the progress rate is of ultimate importance in the ES theory. The value of φ depends on the fitness function analyzed and on the ES algorithm used. Several progress rate formulae can be found in the literature. For the sake of simplicity, only the progress rates for the hyperplane, the sphere model, and the parabolic ridge will be mentioned in this subsection.

5.3.5.1 The hyperplane

The formulae for the progress rate φ are proposed and derived in the literature for different ES algorithms. The progress rate for the $(1, \lambda)$-ES is given in [Rec94, p. 62]. A derivation can be found in [Bey96c, p. 33]. For the (μ, λ)-ES case, the formula is proposed in [Rec94, p. 241], and derived in [Bey96c, p. 187ff]. Similarly, the progress rate for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES is stated in [Rec94, p. 242], and derived in [Bey96c, p. 218]. These three formulae read

$$(1, \lambda)\text{-ES} \quad : \quad \varphi = c_{1,\lambda}\sigma \ , \tag{5.22}$$

$$(\mu, \lambda)\text{-ES} \quad : \quad \varphi = c_{\mu,\lambda}\sigma \ , \tag{5.23}$$

$$(\mu/\mu_{\mathrm{I}}, \lambda)\text{-ES} \quad : \quad \varphi = c_{\mu/\mu,\lambda}\sigma \ . \tag{5.24}$$

Naturally, these values are independent of the number of variables N, and of rotations of the variable axes. The independence of N is specific for this fitness function, whereas the independence of rotations is provided by the isotropic mutation distribution; therefore, it is a property of these ES algorithms.

5.3.5.2 The sphere model

The asymptotically ($N \to \infty$) exact formulae for the progress rate φ of different ES algorithms on the sphere model can be found in the literature. The following normalization [Bey96c, p. 32], [Rec78]

$$\varphi^* := \varphi \frac{N}{D}, \qquad \sigma^* := \sigma \frac{N}{D} \tag{5.25}$$

can be used to make the progress rate formulae stated in [Rec94, p. 64,68,146] independent of the number of variables N and of the residual distance D to the optimum. The N-dependent progress rate formulae are derived in [Bey95a, Bey95b, Bey96b, Bey96c]. The resulting formulae [Bey96c, p. 71,184,212,235] asymptotically ($N \to \infty$) become the same with the ones proposed by Rechenberg without derivation

$$(1,\lambda)\text{-ES} \quad : \qquad \varphi^* = c_{1,\lambda}\sigma^* - \frac{\sigma^{*2}}{2} \ , \tag{5.26}$$

$$(\mu,\lambda)\text{-ES} \quad : \qquad \varphi^* = c_{\mu,\lambda}\sigma^* - \frac{\sigma^{*2}}{2} \ , \tag{5.27}$$

$$(\mu/\mu_{\mathrm{I}},\lambda)\text{-ES} \quad : \qquad \varphi^* = c_{\mu/\mu,\lambda}\sigma^* - \frac{\sigma^{*2}}{2\mu} \ , \tag{5.28}$$

$$(\mu/\mu_{\mathrm{D}},\lambda)\text{-ES} \quad : \qquad \varphi^* = \sqrt{\mu}\, c_{\mu/\mu,\lambda}\sigma^* - \frac{\sigma^{*2}}{2} \ . \tag{5.29}$$

These progress rate formulae can be used for deriving the expected average residual distance $D^{(\infty)}$ at the stationary case without self-adaptation ($\sigma = \text{const}$). By equating $\varphi^* \overset{!}{=} 0$, and using the normalization for σ^* in (5.25), one obtains the following $D^{(\infty)}$ values:

$$(1,\lambda)\text{-ES} \quad : \qquad D^{(\infty)} = \frac{\sigma N}{2c_{1,\lambda}} \ , \tag{5.30}$$

$$(\mu,\lambda)\text{-ES} \quad : \qquad D^{(\infty)} = \frac{\sigma N}{2c_{\mu,\lambda}} \ , \tag{5.31}$$

$$(\mu/\mu_{\mathrm{I}},\lambda)\text{-ES} \quad : \qquad D^{(\infty)} = \frac{\sigma N}{2\mu c_{\mu/\mu,\lambda}} \ , \tag{5.32}$$

$$(\mu/\mu_{\mathrm{D}},\lambda)\text{-ES} \quad : \qquad D^{(\infty)} = \frac{\sigma N}{2\sqrt{\mu} c_{\mu/\mu,\lambda}} \ . \tag{5.33}$$

These formulae will be used in the approximate progress rate formulae as well as in the crude estimation of the stationary distance $R^{(\infty)}$ of ridge functions.

5.3.5.3 The parabolic ridge

The ridge function family has not been analyzed as intensely as the sphere model in the literature. Some analytical formulae are proposed by Rechenberg. In the following, these results and statements in [Rec94, p. 65-66,76-78,214] will be summarized. The paragraphs below mostly consist of the translations from this book.

Rechenberg defines the parabolic ridge using the object variable vector $\mathbf{y}$ as

$$Q = Q_0 + cy_1 - d\sum_{k=2}^{N} y_k^2 \ . \tag{5.34}$$

This definition is equivalent to (3.16). The constant Q_0 does not affect the behavior of the ES algorithms. Similarly, one may divide Q by c, and only the quality gain formulae will be affected. One may choose $Q_0 = 0$ and $c = 1$ to ease the comparison of these two equations. The progress rate φ and the success probability P_{s1} of the $(1, \lambda)$-ES are given by Rechenberg as (on the ridge axis and for $N \gg 1$)

$$\varphi = c_{1,\lambda}\sigma - \frac{Nd\sigma^2}{c} \ , \tag{5.35}$$

$$P_{s1} = \frac{1}{2}\left[1 - \operatorname{erf}\left(\frac{Nd\sigma}{\sqrt{2}c}\right)\right] \ . \tag{5.36}$$

Using Equation (5.7), one gets for $c = 1$

$$\varphi = c_{1,\lambda}\sigma - Nd\sigma^2, \qquad P_{s1} = \Phi(-Nd\sigma). \tag{5.37}$$

Actually, Rechenberg did not formally define the distance r to the ridge axis (Equation (3.12)). He does not give any formulae for $r \neq 0$, and does not suggest any normalization scheme for the parabolic ridge.

In his work [Rec94, p. 65-66,76-78,214], he states that the ES algorithm will follow the ridge axis in the long run. This can be interpreted as "the expected distance r to the ridge axis should be low". Actually, no expected value for r is mentioned in his work. He adds that the progress in the fitness space does not necessarily yield the progress in the search space, and that a general progress rate theory of the ES should still be devised so that it will also be appropriate for the ridge functions.

Based on the smallness of the curvature radius on the ridge axis, Rechenberg asserts that the progress rate φ is small on the ridge axis. However, he adds, the local gradients at the states other than the ones on the ridge axis are directed toward the axis, where the progress rate is smaller. He gives the optimal mutation strength on the ridge axis as

$$\hat{\sigma} = c_{1,\lambda}c/2d\sqrt{N} \ . \tag{5.38}$$

In all fitness landscapes, the aim is to increase the (global) progress rate of the ES algorithm applied. For such special landscapes similar to the parabolic ridge, Rechenberg proposes to increase the mutation strength beyond the values advised by the evolution window in order to reach this goal. Using pictures, he asserts that the length of a mutation vector can be much larger than the distance to the ridge axis.

As to the multiplicative self-adaptation rules (MSR), he notes that these are inappropriate for finding the optimal mutation strength (in the ridge case). For the explanation of this observation he uses the principle of the MSR method: Since the MSR method uses only the (fitness) information of a single generation, it is expected to maximize the quality gain $\overline{Q}$ by using the self-adaptation operator. In order to obtain long term progress, he advises to hold the mutation strength constant for several generations. If one would operate different σ values on different populations, one could compare the results after an isolation period. As he stated, this scheme leads to the hierarchical ES.

According to Rechenberg, the hierarchical ES is the only way to maximize the (global) progress rate φ. Furthermore, he advises the parabolic ridge as a test function for genetic algorithms. Actually, the rotated ridge function (3.17) can also be used by other evolutionary algorithms. The values obtained for the convergence measures can be used to compare the performances of evolutionary algorithms with each other.

5.3.6 The quality gain $\overline{Q}$

In [Bey94], [Bey96c, Chapter 4], the quality gain formulae are derived for the $(1 \overset{+}{,} \lambda)$-ES. These two formulae can also be used for mutation distributions other than the normal distribution. Moreover, a search space other than $\mathbb{R}^N$ can also be used for the fitness function. For example, the quality gain $\overline{Q}$ of the `OneMax` fitness function (3.22) is derived in [Bey96c, pp. 125-128].

In scope of this work, only the search space $\mathbb{R}^N$ is investigated; and the normal distribution is used to generate mutations. The derivation of $\overline{Q}$ is explained here under these conditions. An explanation of this case can be found in [OBS97], an overview of the derivation will be given below.

The derivation of the quality gain formulae is based on the approximation of the local quality function (LQF) $Q(\mathbf{z})$ (Subsection 5.3.2). The $Q(\mathbf{z})$ values are scalar random quantities. In general, they are *not* expected to be normally distributed, even if the mutations on the search space are generated using the normal distribution.

In the final quality gain formulae for the $(1 \overset{+}{,} \lambda)$-ES, only certain statistical parameters describing the $Q(\mathbf{z})$ distribution remain as unknowns. This is also true for the $\overline{Q}$ formula of the $(1+\lambda)$-ES; however, this formula is not investigated in this work. The progress coefficients $c_{1,\lambda}$, $d^{(2)}_{1,\lambda}$, and $d^{(3)}_{1,\lambda}$ –which also occur in the $\overline{Q}$ formula– were introduced in Subsection 5.3.4. Their values can be read from tables [Bey96b], [Bey96c, p. 351], or computed using numerical integration tools (such as Mathematica).

For a given state $\mathbf{x}$ in the search space, the local quality function $Q(\mathbf{z})$ describes the scalar fitness values for any $\mathbf{z}$. Because of the approximation of $Q(\mathbf{z})$ in (5.15) using the vector $\mathbf{a}$ and matrix $\mathbf{Q}$, the $Q(\mathbf{z})$ values are accurate for sufficiently small mutation strengths. As a result, the $\overline{Q}$ formula is expected to be less accurate for ridge functions with $\alpha \neq 2$, i.e. if higher order derivatives are not negligible. For $\alpha = 2$ (and for the trivial case $\alpha = 0$), the $Q(\mathbf{z})$ approximation is exact.

Assume that the random variable Q describes the $Q(\mathbf{z})$ values. The ultimate aim of the derivation of the $\overline{Q}$ formula is the determination of the probability density function (pdf) of Q. The mean value M_Q and the standard derivation S_Q of this pdf can easily be computed. In the next step, the pdf of Q is *standardized* so that it has the mean zero and variance one, yielding the standardized variable $z := (Q - M_Q)/S_Q$. Even after this standardization, the pdf does not become a normal distribution. The mean value and variance of the pdf of z match to the ones of $\mathcal{N}(0,1)$; however, the higher order moments differ. This difference in the normal estimation can be corrected using Hermite polynomials (Subsection 5.3.1, (5.13)). After this correction, the cumulative distribution function of the standardized variable z was given in Subsection 5.3.3 using the cumulants κ_k. The cumulants emerge as a consequence of the difference in the higher order moments to the normal distribution. In the last step of the quality gain derivation, the quantile density $P_z^{-1}(f)$, i.e. the inverse function of $P_z(z)$, is approximated using a power series of $\Phi^{-1}(f)$ (quantile function of the normal distribution). Using this approximation for the cumulative distribution, and the progress coefficients introduced in Subsection 5.3.4, the quality gain formula is derived for the $(1,\lambda)$-ES [Bey94], [Bey96c, p. 118]

$$\overline{Q} = M_Q + S_Q\left\{ -\frac{\kappa_3}{6} + c_{1,\lambda}\left(1 + \frac{5}{36}\kappa_3^2 - \frac{\kappa_4}{8}\right) + d_{1,\lambda}^{(2)}\frac{\kappa_3}{6} + d_{1,\lambda}^{(3)}\left(\frac{\kappa_4}{24} - \frac{\kappa_3^2}{18}\right) + \dots\right\}. \quad (5.39)$$

The calculation of the $\overline{Q}$ value for the $(1,\lambda)$-ES is a technical task. The values of the progress coefficients can be obtained numerically or read from tables. The values of the parameters M_Q, S_Q, κ_3, and κ_4 depend on the fitness function to be analyzed and mutation distribution used. The formulae for correlated mutations can be found in [Bey96c, pp. 120-122]. The case of isotropic normal mutations is considered in this work, and the relevant formulae can be found below. In these formulae, the notation q_i is used for the diagonal entries $(Q)_{ii}$ of the matrix $\mathbf{Q}$ (Equation (5.16)). The *trace* of the matrix $\mathbf{Q}$, i.e. the sum of its diagonal entries, is denoted by $\mathrm{Tr}[\mathbf{Q}] := \sum_{i=1}^{N} q_i$.

The complete Q matrix. If the off-diagonal entries of the $\mathbf{Q}$ matrix (see Equation (5.16)) are nonzero, then one has to use the following formulae to describe the local quality distribution [Bey94], [Bey96c, p. 123]:

$$M_Q = -\sigma^2\mathrm{Tr}[\mathbf{Q}], \qquad S_Q = \sigma\sqrt{\|\mathbf{a}\|^2 + 2\sigma^2\mathrm{Tr}[\mathbf{Q}^2]}, \quad (5.40)$$

$$\kappa_3 = -\frac{\sigma^4}{S_Q^3}\left(6\mathbf{a}^{\mathrm{T}}\mathbf{Q}\mathbf{a} + 8\sigma^2\mathrm{Tr}[\mathbf{Q}^3]\right), \qquad \kappa_4 = \frac{48\sigma^6}{S_Q^4}\left(\|\mathbf{Q}\mathbf{a}\|^2 + \sigma^2\mathrm{Tr}[\mathbf{Q}^4]\right). \quad (5.41)$$

The value of $\|\mathbf{a}\|^2 = \sum_{i=1}^{N} a_i^2$ can be computed easily, but the calculation of $\mathrm{Tr}[\mathbf{Q}^2]$ requires more effort. As a result, one can at least compute M_Q and S_Q with a moderate effort for any fitness function. As explained in Subsection 5.3.3, one can obtain a first order approximation for the success probability as $P_{s1} \approx \Phi(M_Q/S_Q)$. Therefore, a first estimate of the success probability P_{s1} will be possible. The calculation of $\mathrm{Tr}[\mathbf{Q}^3]$ and $\mathrm{Tr}[\mathbf{Q}^4]$ is very lengthy, but mathematically it is a trivial task.

The diagonal Q matrix. If all off-diagonal entries of $\mathbf{Q}$ are zero, the computation of these four quantities is much simpler. This is for example the case for the parabolic ridge, hyperplane, and the member (3.7) of the sphere model family:

$$M_Q = -\sigma^2 \sum_{i=1}^{N} q_i, \qquad S_Q = \sigma \sqrt{\|\mathbf{a}\|^2 + 2\sigma^2 \sum_{i=1}^{N} q_i^2}\ , \tag{5.42}$$

$$\kappa_3 = -\frac{\sigma^4}{S_Q^3}\left(6\sum_{i=1}^{N} a_i^2 q_i + 8\sigma^2 \sum_{i=1}^{N} q_i^3\right), \qquad \kappa_4 = \frac{48\sigma^6}{S_Q^4}\left(\sum_{i=1}^{N} a_i^2 q_i^2 + \sigma^2 \sum_{i=1}^{N} q_i^4\right). \tag{5.43}$$

5.3.7 Induced order statistics

As compared to the calculation of the quality gain $\overline{Q}$, the calculation of the progress rate φ poses several analytical difficulties. An overview will be given here for the $(1,\lambda)$-ES case. This method was developed by Beyer for the progress rate analysis of ES algorithms. It is called "induced order statistics" and occurs in almost all progress rate derivations in [Bey96c].

The progress rate φ was defined as the expected value of the decrease in the distance to the optimum (Subsection 4.1.2). In other words, φ gives the effective distance traveled toward the optimum in a single generation. If one denotes this distance by the random variable z, and its pdf by $p_{1,\lambda}(z)$, respectively, the progress rate of the $(1,\lambda)$-ES can be expressed using the definition of the expected value

$$\varphi = \mathrm{E}\{z\} = \int_{-\infty}^{\infty} z\, p_{1,\lambda}(z)\,\mathrm{d}z\ . \tag{5.44}$$

The integral is taken over all possible values of z: The comma selection strategy accepts worsenings; therefore, z can have all values of $\mathbb{R}$. For the plus strategy, an additional pdf should be multiplied by $p_{1,\lambda}(z)$ to obtain $p_{1+\lambda}(z)$ (See [Bey96c, p. 82]). This additional pdf reflects the condition that the best offspring should be better than the parent. Alternatively, the integration limits are to be changed appropriately.

Since isotropic mutations are used in generating the descendants, the mutations have the expected value zero and the variance σ^2 around the parent, in any direction of the search space. Therefore, also the random variable z is generated according to this distribution, denoted by $p_z(z) \sim \mathcal{N}(0,\sigma^2)$ (See Equation 5.4). However, in order to be selected as the best individual, a descendant having the pdf $p_z(z)$ must have a better fitness than the other $\lambda-1$ descendants. This condition is reflected by the cumulative distribution of acceptance, $P_{a\,1,\lambda}(z)$. Since any of the λ descendants can principally have the best fitness value, there are λ different constellations. Therefore, $p_{1,\lambda}(z)$ can be further specified as

$$p_{1,\lambda}(z) = \lambda\, p_z(z)\, P_{a\,1,\lambda}(z)\ . \tag{5.45}$$

The probability distribution $P_{a\,1,\lambda}(z)$ states the probability that $\lambda - 1$ descendants have fitness values that are worse than the fitness value of the individual with a given z. For this individual, the probability distribution of the local quality function (LQF) conditional to a given z is denoted by $p(Q_{|z}|z)$. Therefore, $Q_{|z}$ denotes the conditional random variable specifying the LQF value for a given z. Let $P_1(Q_{|z})$ denote the cumulative distribution that a single descendant out of $\lambda - 1$ others has a LQF value less than $Q_{|z}$. Consequently, $[P_1(Q_{|z})]^{\lambda-1}$ states the probability that *all* $\lambda - 1$ descendants have LQF values less than $Q_{|z}$. As a result, the $P_{a\,1,\lambda}(z)$ value can be stated as

$$P_{a\,1,\lambda}(z) = \int_{-\infty}^{\infty} p(Q_{|z}|z)\,[P_1(Q_{|z})]^{\lambda-1}\,\mathrm{d}Q_{|z} \ . \tag{5.46}$$

The distribution $P_1(Q_{|z})$ can be specified further. It is obtained as the probability distribution which gives the probability of all possible LQF values worse than $Q_{|z}$ for all possible z values. This can be expressed as a double integral. The inner integral is taken for all possible z values, the outer one from the worst possible LQF value (that is, practically $-\infty$) up to $Q_{|z}$. The density $p(Q_{|z}|z)$ will be approximated using the normal distribution. If the integral boundaries are finite, they could be extended to $-\infty$ and ∞ because of an interesting property of the normal distribution: The normal distribution is massively concentrated around its mean. For example, the distribution $\mathcal{N}(0, \sigma^2)$ has 68.3% of its density in the interval $[-\sigma, \sigma]$, 95.5% in $[-2\sigma, 2\sigma]$, and 99.7% in $[-3\sigma, 3\sigma]$, respectively. Therefore, the integration limits can be extended to $\pm\infty$, respectively, by accepting relatively negligible errors. This extension yields integrals which can be treated analytically, or at least partially.

Since the inner integral is taken for all possible z values, the result is no more dependent on z. Therefore, this distribution is denoted by $P_1(Q)$ in the following. One obtains

$$\begin{aligned} P_1(Q) &= \int_{-\infty}^{Q}\int_{-\infty}^{\infty} p(Q_{|z}|z)\,p_z(z)\,\mathrm{d}z\,\mathrm{d}Q_{|z} &(5.47)\\ &= \int_{-\infty}^{\infty} p_z(z)\int_{-\infty}^{Q} p(Q_{|z}|z)\,\mathrm{d}Q_{|z}\,\mathrm{d}z \ . &(5.48)\end{aligned}$$

The exchange of the integration order gives (5.48). Note that the density $p_z(z)$ was already used in (5.45) as the distribution that generated the mutation z.

The equations (5.44), (5.45), and (5.46) can now be combined together to yield

$$\varphi_{1,\lambda} = \lambda\int_{-\infty}^{\infty} z\,p_z(z)\int_{-\infty}^{\infty} p(Q_{|z}|z)\,\left[P_1(Q_{|z})\right]^{\lambda-1}\,\mathrm{d}Q_{|z}\,\mathrm{d}z \ . \tag{5.49}$$

Since $P_1(Q)$ is independent of z, the integration order can be exchanged in order to simplify the integration. The substitution $Q := Q_{|z}$ for the parameter of $P_1(Q)$ underlines the independence of $P_1(Q)$ from z. One obtains for the progress rate φ

$$\varphi_{1,\lambda} = \lambda\int_{-\infty}^{\infty} [P_1(Q)]^{\lambda-1}\int_{-\infty}^{\infty} z\,p_z(z)\,p(Q|z)\,\mathrm{d}z\,\mathrm{d}Q \ . \tag{5.50}$$

The definition of the progress rate φ in (5.50) is a *four-tuple* integral (see also the definition of $P_1(Q)$ in (5.47)). In the scope of this work, the density $p(Q|z)$ will be approximated by a normal distribution. As a result, the integral expression for $P_1(Q)$ in (5.48) becomes manageable by using the integral expressions given in Subsection 5.3.1. Actually, the distribution $P_1(Q)$ is given in (5.17), after normalizing the random variable Q. On this approximation level, the cumulants will be neglected, and the cumulative distribution will be approximated by

$$P_1(Q) \approx \Phi\left[\frac{Q - M_Q}{S_Q}\right] \quad . \tag{5.51}$$

The normal approximation of $p(Q|z)$ and the approximation for $P_1(Q)$ cause a negligible error if λ is small as compared to N, as to be shown by simulations. The formula (5.50) is derived for the asymptotics ($N \to \infty$); however, one obtains satisfactory results even for $N \gtrapprox 30$.

For the (μ, λ)-ES, this technique is not sufficient. Since the mutations are practically generated from μ different states, the offspring are not normally distributed in the search space. In [Bey95b], [Bey96c, Chapter 5], a correction term for the skewness of the offspring distribution is introduced. The derivation of $\varphi_{\mu,\lambda}$ is very lengthy for the sphere model. Such an additional approach is not considered in the scope of this work.

Chapter 6

Theory

This chapter consists of results obtained by analyzing ES algorithms on ridge functions. It contains four sections on the respective convergence measures. The ES algorithms, the ridge function family, convergence measures, and the analysis methods were introduced in the four previous chapters, respectively. The results here are organized in sections with respect to the convergence measures of interest. The sequence of sections reflect also the degree of complexity of the analysis, where the most challenging results can be found in the last section. Conversely, some intermediate results in earlier sections are used in deriving some results in the following sections.

The quality gain $\overline{Q}$ is analyzed in Section 6.1. It is a progress measure in the fitness space. Two alternative approaches will be given for the derivation. The results are obtained for the $(1, \lambda)$-ES on the parabolic ridge; whereas both approaches are also applicable to other ridge functions. In Section 6.2, the success measures P_{s1} and $P_{s\lambda}$ will be derived for the $(1, \lambda)$-ES on ridge functions. The relation between the stationary P_{s1} formulae of the sphere model and parabolic ridge is obtained using the asymptotic limit $(\sigma \to \infty)$.

Section 6.3 is the most challenging section of this chapter. The progress rate φ will be derived for several ES algorithms on the general case of ridge functions. The results for the stationary case will additionally be obtained using a local model. For the static case, the method called "*induced order statistics*" will be applied to the $(1, \lambda)$-ES and to the $(\mu/\mu_I, \lambda)$-ES. The latter result will be generalized to the $(\mu/\mu_D, \lambda)$-ES. For the (μ, λ)-ES, the respective formula is obtained by a simple heuristic reasoning.

The distance r to the ridge axis will be analyzed in Section 6.4 on the parabolic ridge. The state equation obtained for the $(1, \lambda)$-ES will be used to calculate the stationary $R^{(\infty)}$ value, the progress measure φ_R for the alternative $\overline{Q}$ formula, and the time constant for a given $r^{(0)}$ value. A similar state equation will be obtained analytically for the $(\mu/\mu_I, \lambda)$-ES, and the corresponding $R^{(\infty)}$ value will be derived. For the $(\mu/\mu_D, \lambda)$-ES, this stationary value will be calculated using the relationship to the $(\mu/\mu_I, \lambda)$-ES. It will be obtained for the (μ, λ)-ES by reasoning on other $R^{(\infty)}$ formulae.

The analysis will be carried out for isotropic mutations and for the asymptotic case $(N \to \infty)$. However, the results can be extended to finite N as long as the condition $N \gg \lambda$ holds, i.e. if λ is constant, or small as compared to N. It will be assumed that the fitness values are obtained without perturbation of noise. The theoretical results will be compared with experiments in the next chapter (Chapter 7).

A small notice on the notation should be added here. The static formulae derived in this chapter for the theoretical quantities φ, $\overline{Q}$, P_{s1}, etc. contain r as a variable. For the stationary case, the stationary value for r is only available for the parabolic ridge. Therefore, in all other cases, r should be treated as an independent variable. Strictly speaking, this fact should be reflected in the notation. For instance, the notation $\varphi_{|r}$ should be used to symbolize the progress rate formula containing r as a variable. This notation would successfully reflect that the progress rate formula is conditional to r, which is an unknown quantity. The symbol φ should be reserved for the stationary progress rate with a known $R^{(\infty)}$, which is already inserted to the formula. Furthermore, the dependence on r should also be reflected on the quantities used in derivations if they contain r. Since such formal notations make the derivations less readable, they are omitted. The static formulae always depend on r, the $R^{(\infty)}$ values are inserted in the stationary ones if they were available.

6.1 The quality gain $\overline{Q}$

This section is dedicated to the calculation of quality gain values for the $(1, \lambda)$-ES. The $\overline{Q}$ formula for the parabolic ridge will be derived. Additionally, another approach for calculating the $\overline{Q}$ value of the parabolic ridge will be given, based on the expected values of progress measures in the search space. Using these two approaches, the $\overline{Q}$ formula for the general ridge function can also be derived.

The quality gain gives the same values as the progress rate φ *if* some conditions are satisfied. Two of such conditions will be stated in this section. Furthermore, the local quality function $Q(\mathbf{z})$ will be derived for the general case of ridge functions. Two function-dependent parameters of $\overline{Q}$, namely the mean and the variance of $Q(\mathbf{z})$, will be calculated for the general case, too.

6.1.1 The local quality function $Q(\mathbf{z})$

The local quality function was introduced in (4.3) for further formalizing the quality gain $\overline{Q}$. The quantity $Q(\mathbf{z})$ was approximated using a Taylor series in Subsection 5.3.2 on Page 60. In this approximation of $Q(\mathbf{z})$, the vector $\mathbf{a}$ and the matrix $\mathbf{Q}$ were introduced (cf. Equations (5.15) and (5.16)). As already mentioned, this approximation is exact if all third and higher order derivatives of the fitness function vanish at the given state. This is for example the case for the hyperplane test function and for the parabolic ridge $F_9(\mathbf{x})$ in (3.16).

In this subsection, the values of $\mathbf{a}$ and $\mathbf{Q}$ will be computed for the general ridge function $F_R(\mathbf{x})$ in (3.11) on Page 32, and for the rotated general case of the hyperplane $F_{hp}(\mathbf{x})$ in (3.21) on Page 34.

6.1.1.1 For ridge functions

The components of the vector $\mathbf{a}$ and the entries in the matrix $\mathbf{Q}$ will be computed next for the general case of ridge functions. For the vector $\mathbf{a}$, they consist of first order partial derivatives, whereas the entries of $\mathbf{Q}$ are obtained by partially differentiating the fitness function with respect to the two corresponding variables. Hence applying the definitions in (5.16) to the definition of $F_R(\mathbf{x})$ in (3.11), and using the definition of the distance r to the ridge axis in (3.12), one obtains

$$(\mathbf{a})_i := \frac{\partial F}{\partial x_i} \quad = \quad \begin{cases} 1 & \text{for } i = 0 \\ -d\alpha r^{\alpha-2} x_i & \text{otherwise,} \end{cases} \tag{6.1}$$

and

$$(\mathbf{Q})_{ij} := -\frac{1}{2}\frac{\partial^2 F}{\partial x_i \partial x_j} \quad = \quad \begin{cases} 0 & \text{for } i = 0 \text{ or } j = 0 \\ \frac{1}{2} d\alpha r^{\alpha-4}[(\alpha-2)x_i^2 + r^2] & \text{for } i = j \text{ and } i \neq 0 \\ \frac{1}{2} d\alpha(\alpha-2) r^{\alpha-4} x_i x_j & \text{otherwise.} \end{cases} \tag{6.2}$$

These formulae get simpler for the parabolic ridge $F_9(\mathbf{x})$ in (3.16), i.e. for $\alpha = 2$. The results can be written shortly as

$$\mathbf{a} = \begin{pmatrix} 1 \\ -2dx_1 \\ -2dx_2 \\ \vdots \\ -2dx_{N-1} \end{pmatrix}, \qquad \mathbf{Q} = \begin{pmatrix} 0 & & & & \mathbf{0} \\ & d & & & \\ & & d & & \\ & & & \ddots & \\ \mathbf{0} & & & & d \end{pmatrix}. \tag{6.3}$$

6.1.1.2 For the rotated hyperplane

This function was defined as $F_{hp}(\mathbf{x})$ in (3.21). The components of vector $\mathbf{v}$ occur in the definition of $\mathbf{a}$, and the matrix $\mathbf{Q}$ is simply the null matrix. The application of (5.16) on (3.21) yields

$$\mathbf{a} = c \cdot \begin{pmatrix} v_0 \\ v_1 \\ v_2 \\ \vdots \\ v_{N-1} \end{pmatrix}, \qquad \mathbf{Q} = \mathbf{0}. \tag{6.4}$$

The variables are numbered starting from zero in order to simplify the comparison with ridge functions. For $\mathbf{v} = (1, 0, \ldots, 0)^{\mathrm{T}}$ and $c = 1$, the same result can be read in (6.1) and (6.2) for $\alpha = 0$.

6.1.2 The pdf of $Q(\mathbf{z})$

The quality gain formula for the $(1,\lambda)$-ES was stated in (5.39). In this formula, the parameters M_Q, S_Q, κ_3, and κ_4 depend on the fitness function of interest. These parameters stand for the moments of the pdf $Q(\mathbf{z})$. They will be computed here using the values for the vector $\mathbf{a}$ and the matrix $\mathbf{Q}$ given in the previous subsection. For the general ridge function $F_R(\mathbf{x})$, only the values of M_Q and S_Q will be given.

6.1.2.1 The rotated hyperplane

The values of $\mathbf{a}$ and $\mathbf{Q}$ can be obtained from (6.4). Since the null matrix $\mathbf{Q} = \mathbf{0}$ is a special case of diagonal matrices, the formulae in (5.42, 5.43) can be used. As a result, one immediately obtains

$$M_Q = 0, \quad S_Q = c\sigma, \quad \kappa_3 = 0, \quad \kappa_4 = 0 \ . \tag{6.5}$$

The equality $\|\mathbf{v}\| = 1$ is used in the calculation of S_Q.

6.1.2.2 The parabolic ridge $F_9(\mathbf{x})$

The four above-mentioned parameters will be computed here for the fitness function $F_9(\mathbf{x})$ in (3.16). The same calculation becomes very complicated for the rotated case, since the $\mathbf{Q}$ matrix will not be diagonal anymore. Using the values of $\mathbf{a}$ and $\mathbf{Q}$ in (6.3), the parameters are obtained here using (5.42) and (5.43).

Some important notes will help the reader in the recalculation of these parameters. Firstly, it is essential to remember that the variables are numbered for ridge functions from zero up to $N-1$. Therefore, this fact should be considered in the application of the parameter definitions. Additionally, the following equalities are used in the calculations:

$$\sum_{i=0}^{N-1} q_i^k = (N-1)d^k, \qquad k \in \mathbb{N} \tag{6.6}$$

$$\|\mathbf{a}\|^2 = \sum_{i=0}^{N-1} a_i^2 = 1 + 4d^2 \sum_{i=1}^{N-1} x_i^2 = 1 + (2dr)^2 \tag{6.7}$$

The definition of r was given in (3.12). Using the intermediate results in (6.6) and (6.7), the parameters read

$$M_Q = -(N-1)d\sigma^2 \qquad S_Q = \sigma\sqrt{1 + (2dr)^2 + 2d^2(N-1)\sigma^2} \ , \tag{6.8}$$

$$\kappa_3 = -\frac{\sigma^4}{S_Q^3}\left(6d(2dr)^2 + 8d^3(N-1)\sigma^2\right) \qquad \kappa_4 = \frac{48\sigma^6}{S_Q^4}\left(d^2(2dr)^2 + d^4(N-1)\sigma^2\right) \ . \tag{6.9}$$

6.1.2.3 The general ridge function $F_R(\mathbf{x})$

Similarly, the parameters for the quality gain $\overline{Q}$ can also be computed for the general ridge function $F_R(\mathbf{x})$ in (3.11). However, since the $\mathbf{Q}$ matrix is not diagonal, one has to use the general definitions in (5.40) and (5.41) to derive these parameters. Therefore, the calculation of these become quite more difficult. The difficulty is mainly caused by the calculation of $\mathrm{Tr}[\mathbf{Q}^2]$, $\mathrm{Tr}[\mathbf{Q}^3]$, and $\mathrm{Tr}[\mathbf{Q}^4]$.

In scope of this work, only M_Q and S_Q will be computed. Therefore, the quantities $\|\mathbf{a}\|^2$, $\mathrm{Tr}[\mathbf{Q}]$, and $\mathrm{Tr}[\mathbf{Q}^2]$ must be derived. After combining these partial results, M_Q and S_Q will be obtained. They will be used in the first order approximation of the success probability P_{s1} (cf. Subsection 5.3.3) and in the derivation of the progress rate formulae.

The quantity $\|\mathbf{a}\|^2$ is obtained in a straightforward manner. Starting at the definition of $\mathbf{a}$ in (6.1), and using the definition of r (3.12) in the last step, one obtains

$$\|\mathbf{a}\|^2 = \sum_{i=0}^{N-1} \left(\frac{\partial F}{\partial x_i}\right)^2 = 1 + \left(d\alpha r^{\alpha-2}\right)^2 \sum_{i=1}^{N-1} x_i^2 = 1 + \left(d\alpha r^{\alpha-1}\right)^2. \tag{6.10}$$

The trace of the matrix $\mathbf{Q}$ in (6.2) is determined in a relatively simple manner, by using (3.12) again

$$\begin{aligned} \mathrm{Tr}[\mathbf{Q}] &= -\frac{1}{2}\sum_{i=0}^{N-1} \frac{\partial^2 F}{\partial x_i^2} = \frac{1}{2} d\alpha r^{\alpha-4} \left[(N-1)r^2 + (\alpha-2)\sum_{i=1}^{N-1} x_i^2\right] \\ &= \frac{1}{2} d(N+\alpha-3)\alpha r^{\alpha-2} \ . \end{aligned} \tag{6.11}$$

Unfortunately, $\mathrm{Tr}[\mathbf{Q}^2]$ is not as easy to compute as $\mathrm{Tr}[\mathbf{Q}]$. First of all, at least the diagonal of matrix $\mathbf{Q}^2$ should be obtained by matrix multiplication. For the calculation of $\mathrm{Tr}[\mathbf{Q}^2]$ for matrix $\mathbf{Q}$ in (6.2),

$$\mathrm{Tr}[\mathbf{Q}^2] := \sum_{i=0}^{N-1} (Q^2)_{i\,i} = \sum_{i=1}^{N-1}\sum_{k=1}^{N-1} (Q)_{i\,k}(Q)_{k\,i} \ , \tag{6.12}$$

the first row and the first column of $\mathbf{Q}$ is ignored since their entries are zero. One can introduce the substitutions

$$A := \frac{1}{2} d\alpha(\alpha-2) r^{\alpha-4}, \qquad B := \frac{1}{2} d\alpha r^{\alpha-2} \ , \tag{6.13}$$

in order to simplify the definitions of the matrix $\mathbf{Q}$ entries in (6.2) to

$$(Q)_{i\,i} = A x_i^2 + B, \qquad (Q)_{i\,k} = (Q)_{k\,i} = A x_i x_k \ . \tag{6.14}$$

After this simplification, the values in (6.14) can be inserted in (6.12) in order to compute the inner sum

$$\begin{aligned}\sum_{k=1}^{N-1}(Q)_{ik}(Q)_{ki} &= (Q)_{ii}^2 + \sum_{\substack{k=1\\k\neq i}}^{N-1}(Q)_{ik}(Q)_{ki} = B^2 + 2ABx_i^2 + A^2x_i^4 + A^2\sum_{\substack{k=1\\k\neq i}}^{N-1}x_i^2x_k^2 \\ &= B^2 + 2ABx_i^2 + A^2x_i^2\sum_{k=1}^{N-1}x_k^2 = B^2 + 2ABx_i^2 + A^2r^2x_i^2 \ . \end{aligned} \tag{6.15}$$

Considering (3.12), (6.13), (6.15), and $A = (\alpha-2)B/r^2$, the calculation of the outer sum in (6.12) will give us the desired value

$$\begin{aligned}\mathrm{Tr}[\mathbf{Q}^2] &= \sum_{i=1}^{N-1}(B^2 + 2ABx_i^2 + A^2r^2x_i^2) = (N-1)B^2 + (2AB + A^2r^2)\sum_{i=1}^{N-1}x_i^2 \\ &= (N-1)B^2 + 2ABr^2 + A^2r^4 = (N-2)B^2 + (B + 2Ar^2)^2 \\ &= (N-2)B^2 + (B + (\alpha-2)B^2)^2 = (N-2)B^2 + (\alpha-1)^2B^2 \\ &= \left[N - 2 + (\alpha-1)^2\right]B^2 = \left[N-2+(\alpha-1)^2\right]\left(\frac{d\alpha}{2}r^{\alpha-2}\right)^2 \ . \end{aligned} \tag{6.16}$$

After the computation of $\|\mathbf{a}\|^2$, $\mathrm{Tr}[\mathbf{Q}]$, and $\mathrm{Tr}[\mathbf{Q}^2]$, the values of M_Q and S_Q can be determined for the general ridge function $F_R(\mathbf{x})$. Please note that these values are valid in scope of the $Q(\mathbf{z})$ approximation introduced in Subsection 5.3.2, since the values of $\mathbf{a}$ and $\mathbf{Q}$ are obtained based on this approximation. Using (5.40) and (6.11), the parameter M_Q reads

$$\boxed{M_Q = -\sigma^2\mathrm{Tr}[\mathbf{Q}] = -\frac{1}{2}d\alpha\sigma^2(N+\alpha-3)r^{\alpha-2} \ .} \tag{6.17}$$

If one inserts zero for α, the M_Q value for the hyperplane in (6.5) is obtained as a special case. Similarly, the M_Q value for the parabolic ridge (Equation (6.8)) is obtained for $\alpha = 2$. The calculation of S_Q will follow next. Using (5.40), (6.10), and (6.16), the parameter S_Q reads

$$\boxed{S_Q = \sigma\sqrt{1 + (d\alpha r^{\alpha-1})^2 + \frac{\sigma^2}{2}\left[N-2+(\alpha-1)^2\right](d\alpha r^{\alpha-2})^2} \ .} \tag{6.18}$$

The special case $\alpha = 2$ gives the S_Q value in (6.8). Similarly, the S_Q value in (6.5) is obtained for $\alpha = 0$ and $c = 1$, as expected. Please note that M_Q (6.17) and S_Q (6.18) are both conditional to r.

At this point, it should be remembered that the mean value and standard deviation of the $Q(\mathbf{z})$ distribution is approximated by M_Q and S_Q, respectively. Therefore, the knowledge of these two quantities yield a first order estimation on this distribution. As mentioned in Subsection 5.3.3, the success probability P_{s1} is defined using M_Q, S_Q, and the cumulants. Therefore, the results obtained here for these parameters will be used in the approximation of the success probability in Section 6.2. Furthermore, these two parameters will be used in Section 6.3 for estimating $Q(\mathbf{z})$ by a normal distribution. In Section 6.4, this estimation will be used for the parabolic ridge case only.

6.1.3 Two $\overline{Q}$ formulae

The quality gain formula for the $(1, \lambda)$-ES was given in (5.39). In the previous subsection, the fitness dependent parameters of the quality gain $\overline{Q}$ are calculated for the parabolic ridge and hyperplane. These parameters must be calculated anew for each fitness function. The remaining constants are fitness-independent progress coefficients. The properties of these two quality gain formulae will be discussed in this subsection. For the rotated hyperplane, the quality gain formula will be compared with the one for the progress rate. For the parabolic ridge case, the order of parameters will be given with respect to the mutation strength σ and number of parameters N.

6.1.3.1 The rotated hyperplane

The quality gain formula for the fitness function $F_{hp}(\mathbf{x})$ in (3.21) is obtained by inserting the values of the respective parameters given in (6.5) into the $\overline{Q}$ definition in (5.39). The result reads

$$\overline{Q} = c \cdot c_{1,\lambda}\sigma \quad . \tag{6.19}$$

Comparing this result with the progress rate formula in (5.22), one observes the relation $\overline{Q} = c\varphi$. Hence, for the hyperplane case, the quality gain and the progress rate can be converted to each other. The proportionality constant c for this relation is used in transforming the expected values of these measures in the fitness space and search space to each other. For the special case $c = 1$, both measures become equal on the hyperplane. In the next subsection (Subsection 6.1.4), we will investigate whether such a relation is possible for the parabolic ridge case.

Please note that the components of the unit vector $\mathbf{v}$ do not occur in the quality gain formula (6.19). This vector indicates the direction of the hyperplane gradient, and it does not occur in the φ formulae of the hyperplane in Point 5.3.5.1, either. As for the φ case, this observation can be explained by the nature of isotropic mutations. This observation for φ will be investigated further for ridge functions by comparing empirical results obtained for randomly selected unit vectors $\mathbf{v}$ (e.g. Subsection 7.2.4 and Subsection 7.2.7). However, for the $\overline{Q}$ case, such a proof of rotation independence is too cumbersome by deriving the parameters for the rotated case of general ridge function. One may still suppose that $\overline{Q}$ is also rotation-independent, as φ is for isotropic mutations. This presumption seems plausible since the isofitness lines are not deformed by rotation.

6.1.3.2 The parabolic ridge

The quality gain formula (5.39) for the parabolic ridge (3.21) is obtained simply by inserting the parameters given in (6.8) and (6.9) into the $\overline{Q}$ definition. The resulting formula is not as simple as the one for the hyperplane case. Therefore, this formula will not be explicitly displayed here. It will be compared with empirical results and to the progress rate formula in the chapter for experiments, Section 7.3. At this point, it is important to summarize the order relations of these parameters with respect to the isotropic mutation strength σ and the number of variables N. The results are summarized in Table 6.1. The parameters S_Q, κ_3, and κ_4 contain the unknown variable r. The value of r may affect these order relations. In the static analysis, the value of r is constant. The order relations with respect to the mutation strength σ are given first. These are obtained from the terms which have the higher order in σ. In the order derivation for the cumulants, please note that S_Q is of $\mathcal{O}(\sigma^2)$. One obtains $\mathcal{O}(\sigma^2)$ for M_Q and S_Q, and $\mathcal{O}(1)$ for κ_3 and κ_4, respectively. Therefore, the cumulants can be neglected relatively. If one considers the asymptotics for N, $N \to \infty$, one obtains $\mathcal{O}(N)$ for M_Q, $\mathcal{O}(\sqrt{N})$ for S_Q, $\mathcal{O}(N^{-1/2})$ for κ_3, and $\mathcal{O}(N^{-1})$ for κ_4. Again, the effect of the cumulants is asymptotically negligible. However, for finite values of σ or N, they should be taken into account. The $\overline{Q}$ formula for finite N will be examined in the chapter for experiments, Subsection 7.3.1.

Table 6.1: The order relations of the parameters in the quality gain $\overline{Q}$ formula in (5.39) for the static and stationary case. The $(1, \lambda)$-ES on the parabolic ridge.

	order in σ				order in N			
	M_Q	S_Q	κ_3	κ_4	M_Q	S_Q	κ_3	κ_4
static	σ^2	σ^2	1	1	N	$\sqrt{N}$	$N^{-1/2}$	N^{-1}
stationary	σ^2	σ^2	1	1	N	N	N^{-1}	N^{-2}

For the stationary case, r is of $\mathcal{O}(\sigma)$ and of $\mathcal{O}(N)$ (cf. the formulae obtained in Subsection 6.4); therefore, it must be considered in order estimations. One obtains the same values for the asymptotics with respect to σ. For $N \to \infty$, one obtains again $\mathcal{O}(N)$ for M_Q, but $\mathcal{O}(N)$ for S_Q, $\mathcal{O}(N^{-1})$ for κ_3, and $\mathcal{O}(N^{-2})$ for κ_4, respectively.

6.1.4 An alternative approach to $\overline{Q}$

The quality gain was defined in Subsection 4.1.1 using the local quality function $Q(\mathbf{z})$. For the $(1, \lambda)$-ES, it was formalized as the expected value of the $Q(\mathbf{z})$ for the best descendant. The resulting formula based on the statistical moments of the offspring distribution can be found in (5.39). For the parabolic ridge, the required parameters are computed in Subsection 6.1.2.

In this subsection, the definition of $Q(\mathbf{z})$ will be used to derive an alternative $\overline{Q}$ formula for the $(1,\lambda)$-ES on the parabolic ridge. The final formula will be based on the expected values of two progress measures (φ and φ_R) in the search space. The first measure is the progress rate φ (4.6). It measures the expected useful distance traveled toward the optimum in one generation. The second one, φ_R, measures the progress in the direction orthogonal to the ridge axis, i.e. in the r direction. After the formal definition of φ_R, the alternative derivation of $\overline{Q}$ will follow.

The ridge functions were introduced in Subsection 3.3.2. For these fitness functions, the minimization of r is defined as the short term goal, and the maximization of x_0 as long term goal. The measure φ_R is introduced here to formalize the progress orthogonal to the ridge axis, i.e. the short term goal. It gives the expected decrease in the distance to the ridge axis in a single generation. For the $(1,\lambda)$-ES, it can be defined as

$$\varphi_R := \mathrm{E}\{r^{(g)} - r^{(g+1)}\} = \mathrm{E}\{\Delta r^{(g)}\} \ . \tag{6.20}$$

One obtains $\varphi_R < 0$ for $r^{(g)} < R^{(\infty)}$ and $\varphi_R > 0$ for $r^{(g)} > R^{(\infty)}$. The symbol $R^{(\infty)}$ was introduced in Page 49 to denote the stationary value of r. For the stationary case ($r^{(g)} \approx R^{(\infty)}$), φ_R is small; and for $r^{(g)} = R^{(\infty)}$ it is zero.

For the formalization of the quality gain $\overline{Q}$ according to (4.4), the fitness values at generations g and $g+1$ should be used

$$F^{(g)} = x_0^{(g)} - d{r^{(g)}}^2 \tag{6.21}$$

$$F^{(g+1)} = x_0^{(g+1)} - d{r^{(g+1)}}^2 = x_0^{(g)} + z_0^{(g)} - d\Big[r^{(g)} - \Delta r^{(g)}\Big]^2 \ . \tag{6.22}$$

The symbol $z_0^{(g)}$ stands for the component of the mutation which generated the best offspring in the progress direction. The definition of $\Delta r^{(g)}$ can be found in (6.20). After inserting (6.21) and (6.22) in (4.3), one obtains for $Q(\mathbf{z})$

$$Q(\mathbf{z}) = F^{(g+1)} - F^{(g)} = z_0^{(g)} + 2dr^{(g)}\Delta r^{(g)} - d(\Delta r^{(g)})^2 \ . \tag{6.23}$$

The quality gain $\overline{Q}$ was defined in (4.4) using the expected value of the local quality function. It reads for the parabolic ridge using (6.23)

$$\overline{Q} := \mathrm{E}\{F^{(g+1)} - F^{(g)}\} = \mathrm{E}\{z_0^{(g)}\} + 2dr^{(g)}\,\mathrm{E}\{\Delta r^{(g)}\} - d\,\mathrm{E}\{(\Delta r^{(g)})^2\} \ . \tag{6.24}$$

As discussed in Subsection 4.1.2, the first term gives φ. Using the definition of φ_R in (6.20), the second term becomes $2dr^{(g)}\varphi_R$. The expression $\mathrm{E}\{(\Delta r^{(g)})^2\}$ will be assumed to be approximately equal to $\Big[\mathrm{E}\{(\Delta r^{(g)})\}\Big]^2$. The error made will be discussed below. Using this assumption, (6.24) becomes

$$\boxed{\overline{Q} = \varphi + 2dr^{(g)}\varphi_R - d\varphi_R^2 + \dots \ .} \tag{6.25}$$

By this equation, the quality gain value can be obtained using the measures in the search space. This relation is also expected to hold for ES algorithms other than the

$(1,\lambda)$-ES, and therefore, it serves as a more general approach than (5.39). The third term in (6.25) is expected to be negligible as compared to the second term for sufficiently large $r^{(g)}$. This formula will be verified empirically in Section 7.3. For $\varphi_R = 0$, one has $\overline{Q} = \varphi$, in other words, for $r = R^{(\infty)}$, the quality gain is expected to be equivalent to the progress rate. Therefore, one can expect $\overline{Q} \approx \varphi$ for the stationary case ($r \approx R^{(\infty)}$). Additional to the case mentioned in (6.19), we observe a further case in (6.25) where this equivalence is expected to hold. This condition can be observed on the static experiment in Subsection 7.3.2. A theoretical formula for φ_R can be found in Point 6.4.1.8, Equation (6.177).

Estimating the error for $\mathrm{E}\{(\Delta r^{(g)})^2\} \approx \left[\mathrm{E}\{(\Delta r^{(g)})\}\right]^2$. The definition of the *variance* will be used to estimate the error made by this assumption. The variance $D^2\{x\}$ of a random variable x is defined as [BS79, p. 704]

$$D^2\{x\} := \mathrm{E}\{x^2\} - [\mathrm{E}\{x\}]^2 \ . \tag{6.26}$$

Therefore, $\mathrm{E}\{x^2\}$ can be approximated by $[\mathrm{E}\{x\}]^2$ if the variance of x is sufficiently small. This estimation would also be valid for the random variable $\Delta r^{(g)}$ under this condition. For the theoretical estimation of this error, one has to consider

$$\begin{aligned}
\mathrm{E}\left\{(\Delta r^{(g)})^2\right\} &= \mathrm{E}\left\{(r^{(g)} - r^{(g+1)})^2\right\} = r^{(g)2} - 2r^{(g)}\,\mathrm{E}\left\{r^{(g+1)}\right\} + \mathrm{E}\left\{r^{(g+1)2}\right\}, && (6.27)\\
\mathrm{E}\left\{\Delta r^{(g)}\right\} &= r^{(g)} - \mathrm{E}\left\{r^{(g+1)}\right\} \ , && (6.28)\\
\left[\mathrm{E}\left\{\Delta r^{(g)}\right\}\right]^2 &= r^{(g)2} - 2r^{(g)}\,\mathrm{E}\left\{r^{(g+1)}\right\} + \left[\mathrm{E}\left\{r^{(g+1)}\right\}\right]^2 \ . && (6.29)
\end{aligned}$$

Equations (6.27) and (6.29) differ only by their third terms from each other. Therefore, one can equivalently investigate whether $\mathrm{E}\{r^{(g+1)2}\} \approx [\mathrm{E}\{r^{(g+1)}\}]^2$ or $\sqrt{\mathrm{E}\{r^{(g+1)2}\}} \approx \mathrm{E}\{r^{(g+1)}\}$ holds. Unfortunately, it was not possible in scope of this work to determine the conditions for this approximate equality.

6.2 The success probability: P_{s1} and $P_{s\lambda}$

The success measures P_{s1} and $P_{s\lambda}$ were introduced in Section 4.2. The former one stands for the probability to generate an offspring with a fitness value not worse than the parent. The latter one gives the same probability for the whole set of λ offspring, i.e. the probability that at least one of the descendants is better than or as good as its parent. They will be computed here for the $(1,\lambda)$-ES on the parabolic ridge first. A reasonable estimate to P_{s1} is given, the error made thereby is calculated. Finally, an approximate formula for P_{s1} of the $(1,\lambda)$-ES is given on the general ridge function.

The P_{s1} formula can be used to derive $P_{s\lambda}$ using Equation (4.10), $P_{s\lambda} = 1 - [1 - P_{s1}]^{\lambda}$. Therefore, the formulae are not explicitly repeated in this section for $P_{s\lambda}$. The values of success measures at the optimal mutation strength $\hat{\sigma}^*$ will be compared with the simulation results in the next chapter, Section 7.4.

6.2.1 The parabolic ridge case

The parameters for the pdf of the offspring distribution were derived in Point 6.1.2.2. Therefore, the calculation of P_{s1} is a simple task. In Subsection 5.3.3, the P_{s1} formula has been given using the local quality function $Q(\mathbf{z})$ as $P_{s1} = 1 - P(Q(\mathbf{z}) < 0)$. Using the standardized variable $z := (Q - M_Q)/S_Q$, the cumulative density function $P_z(z)$ can be used to express P_{s1}. The density itself $P_z(z)$ is expressed by the series (5.17). For the $(1,\lambda)$-ES on the parabolic ridge, all of the parameters required in this equation can be found in Point 6.1.2.2 (except κ_5). The distribution $P_z(z)$ can be approximated by

$$P_z(z) \approx \Phi(z) - \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\frac{1}{2}z^2} \frac{\kappa_3}{3!} \mathrm{He}_2(z) \ . \tag{6.30}$$

Only the first term in the bracket of (5.17) is considered. The Hermite polynomial $\mathrm{He}_2(z)$ can be calculated using (5.13)

$$\mathrm{He}_2(z) = z^2 - 1 \ . \tag{6.31}$$

In the following, the second term in (6.30) will be neglected in the calculations. The error made thereby is asymptotically ($N \to \infty$) negligible; however, for finite N an approximation error should be expected. In Section 7.4 of the next chapter, it will be shown by experiments that also this error is relatively small. The approximate P_{s1} formula is obtained using (4.9), (5.17), and (6.8)

$$P_{s1} = 1 - P(Q(\mathbf{z}) < 0) = 1 - P_z(z)\big|_{Q=0} \approx 1 - \Phi\left(\frac{0 - M_Q}{S_Q}\right) \tag{6.32}$$

$$P_{s1} \approx \Phi\left(\frac{M_Q}{S_Q}\right) = \Phi\left(-\frac{(N-1)d\sigma}{\sqrt{1 + (2dr)^2 + 2d^2(N-1)\sigma^2}}\right) \ . \tag{6.33}$$

This formula will be used in the static analysis. It attains its minimum value for $r = 0$, and its maximum value $\hat{P_{s1}} = \frac{1}{2}$ for $r \to \infty$ (cf. Section 4.2). The value of $P_{s\lambda}$ can be obtained using (4.10). For the stationary case, the $R^{(\infty)}$ value in (6.166) from Section 6.4.1 must be inserted in (6.33). One obtains

$$P_{s1} \approx \Phi \left[-\frac{d(N-1)\sigma}{\sqrt{1 + \frac{[d(N-1)\sigma]^2}{2c_{1,\lambda}^2} \left(1 + \sqrt{1 + \left(\frac{2c_{1,\lambda}}{d(N-1)\sigma} \right)^2} \right)}} \right] . \tag{6.34}$$

This result was obtained for $N \to \infty$ and after approximating the distributions of the Q variates by normal distributions. Please note that P_{s1} is conditional to r in Equation (6.33), but not in Equation (6.34). In (6.34), the term "$d(N-1)\sigma$" occurs three times. As will be seen in the derivation of the normalized progress rate formula in Section 6.3, this term appears to be the appropriate definition for the normalized mutation strength σ^*. The same definition can be obtained by substituting $\alpha = 2$ in (4.11). For $\sigma^* := d(N-1)\sigma$, one gets the normalized stationary P_{s1} value

$$P_{s1} \approx \Phi \left[-\left(\frac{1}{\sigma^{*2}} + \frac{1}{2c_{1,\lambda}^2} \left(1 + \sqrt{1 + \left(\frac{2c_{1,\lambda}}{\sigma^*} \right)^2} \right) \right)^{-\frac{1}{2}} \right] . \tag{6.35}$$

One obtains asymptotically

$$\lim_{\sigma^* \to \infty} P_{s1} = \Phi(-c_{1,\lambda}) . \tag{6.36}$$

This limit and the formulae (6.33), (6.34), and (6.35) are obtained for the asymptotic limit case ($N \to \infty$). However, they also yield accurate results for finite N, especially at the stationary case ($r \approx R^{(\infty)}$). The principal requirement for that is $N \gg \lambda$; otherwise, no accuracy can be guaranteed.

As will be seen in Subsection 6.3.1, the maximum normalized progress rate $\hat{\varphi}^*$ will be obtained on the parabolic ridge as σ^* goes to infinity. Therefore, (6.36) gives the value of P_{s1} at optimal σ^*, i.e. at $\hat{\sigma}^*$. In other words, the maximum progress rate is obtained where the success probability P_{s1} attains its minimum value. This value for P_{s1} differs considerably from the corresponding values for the hyperplane, sphere model, and corridor model [Rec73, p. 122], [Sch95, p. 143], which are 0.5, 0.27, and $1/2e$, respectively (see Page 52). Depending on the value of λ, (6.36) is much smaller than the values obtained for these three functions analyzed in the literature.

The relation to the sphere model. The relation of ridge functions to the sphere model was explained at the end of Subsection 3.3.2. Therefore, it is reasonable to compute the limit value of P_{s1} for $\sigma \to \infty$ on the sphere model, and compare it to the limit in (6.36) on the parabolic ridge.

For the function $F_2(\mathbf{x}) := -D^2$ in (3.7), one can obtain the vector $\mathbf{a}$ and the matrix $\mathbf{Q}$ to approximate the local quality function $Q(\mathbf{z})$. Using (5.16) the result reads

$$\mathbf{a} = \begin{pmatrix} -2x_1 \\ -2x_2 \\ -2x_3 \\ \vdots \\ -2x_N \end{pmatrix}, \qquad \mathbf{Q} = \begin{pmatrix} 1 & & & & \mathbf{0} \\ & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ \mathbf{0} & & & & 1 \end{pmatrix} . \tag{6.37}$$

Using the definition of M_Q and S_Q given in (5.42), the definition of D in (3.6), and (6.32) for approximating P_{s1}, one gets

$$P_{s1} \approx \Phi\left(\frac{M_Q}{S_Q}\right) = \Phi\left(-\frac{N\sigma^2}{\sigma\sqrt{4D^2 + 2N\sigma^2}}\right) . \tag{6.38}$$

Since the limit in (6.36) was obtained for the stationary case, the P_{s1} value will also be evaluated here for the same case. Inserting the $D^{(\infty)}$ value from (5.30) in (6.38), the formula reads

$$P_{s1} \approx \Phi\left(-\frac{N\sigma}{\sqrt{4\left(\frac{N\sigma}{2c_{1,\lambda}}\right)^2 + 2N\sigma^2}}\right) = \Phi\left(-\frac{1}{\sqrt{c_{1,\lambda}^{-2} + 2N^{-1}}}\right) . \tag{6.39}$$

For $N \to \infty$, (6.39) is equivalent to the value in (6.36) since $c_{1,\lambda}^{-2} \gg 2N^{-1}$. In the sphere model case, this limit means also $\varphi^* = 0$, i.e. no progress toward the optimum is expected for the $(1,\lambda)$-ES if $D \approx D^{(\infty)}$. In the parabolic ridge case, this limit corresponds to the maximum progress rate for the stationary case, as will be seen in Subsection 6.3.1.

6.2.2 The general ridge case

In the previous subsection, the P_{s1} formula has been derived for the $(1,\lambda)$-ES on the parabolic ridge. Naturally, similar formulae can be derived for other ES algorithms on the parabolic ridge, or for the $(1,\lambda)$-ES on other ridge functions. The derivation of the P_{s1} formula for the $(1,\lambda)$-ES on the general ridge function will analogously be done as the derivation of (6.33) or (6.38). The required M_Q and S_Q values can be found in (6.17) and (6.18), respectively. Therefore, P_{s1} reads

$$P_{s1} \approx \Phi\left(\frac{M_Q}{S_Q}\right) = \Phi\left(-\frac{d\alpha\sigma(N+\alpha-3)r^{\alpha-2}}{2\sqrt{1 + (d\alpha r^{\alpha-1})^2 + \frac{\sigma^2}{2}\left[N - 2 + (\alpha-1)^2\right](d\alpha r^{\alpha-2})^2}}\right) . \tag{6.40}$$

Naturally, this result reduces for $\alpha = 2$ to the P_{s1} formula for the parabolic ridge in (6.33).

Unfortunately, the estimation of the error made in (6.40) has not been done yet, since the κ_3 parameter is necessary in (6.30) for that. Its computation is lengthy. Equation (6.40) is expected to yield useful results as least for the stationary case (see Subsection 7.4.3).

6.2.3 Final remarks on the success probability

The success measures P_{s1} and $P_{s\lambda}$ are based on the local quality function $Q(\mathbf{z})$. Therefore, it is related directly to the quality gain $\overline{Q}$. Consequently, similar to the quality gain $\overline{Q}$, success measures are also measured in the fitness space. The progress rate φ, however, is measured in the search space. The success rate P_{s1} and the progress rate φ will be used in the static analysis of the parabolic ridge to investigate the relationship between φ and P_{s1} for this fitness function. These empirical results for P_{s1} in Section 7.4 will be compared with the ones in Section 7.2 obtained for the $(1, \lambda)$-ES, yielding interesting observations.

Measures on both fitness and search spaces are concerned in this work. The analysis of measures in the search space will be carried out in the following two sections. The important task thereafter is to speculate on the role of the measures of these two spaces, and on the relative merit obtained from them.

6.3 The progress rate φ

This section is devoted to the analysis of the progress rate on ridge functions. Using a local model, the stationary case will be investigated for the general ridge function on various ES algorithms (Subsection 6.3.1). Thereafter, the $(1, \lambda)$-ES case will be analyzed in Subsection 6.3.2 for the static case. The same analysis will be repeated for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES in Subsection 6.3.3. In Subsection 6.3.4, the results from the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES will be adapted to the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES. For the (μ, λ)-ES, the analytical results from the $(1, \lambda)$-ES will be used to obtain the static φ formula (Subsection 6.3.5). For each static result mentioned, the parabolic ridge case is considered in detail using respective $R^{(\infty)}$ results from Section 6.4. The maximum φ value for these four algorithms on the parabolic ridge will be compared to each other in Subsection 6.3.6, which immediately leads to the notion of progress rate per descendant for fair performance comparisons, namely the *progress efficiency* η. The results of this section are summarized in Subsection 6.3.7.

6.3.1 A local model for the stationary case

Principally, the progress rate values can be computed using the method based on induced order statistics. This method was described in Subsection 5.3.7; and it will be used for the analytical derivation of the progress rate formulae for the $(1, \lambda)$-ES and for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES in the next two subsections. In this section, an alternative method based on geometric relations will be provided. This method is much simpler. It yields satisfactory results for the general case of ridge functions and for a large spectrum of ES algorithms.

This local model is based on two assumptions: The isofitness surface can be locally approximated for the stationary state by a hyperplane and the progress attained can be decomposed in two directions perpendicular to each other. The former precondition -the hyperplane approximation- will prove itself to be valid at least for the mutation strength corresponding to $R^{(\infty)}$. This observation will be verified by comparing the φ formulae obtained in this subsection with the ones obtained in Subsection 6.3.2 and Subsection 6.3.3. This comparison will be done on the formulae themselves to identify the negligible terms for the stationary case, and using simulation results in Section 7.2. The latter statement -decomposition of progress- is validated by the evolutionary progress principle (EPP, Subsection 5.2.6); and it will serve as an interesting example on how the loss term and the gain term may be placed in the progress rate formula.

6.3.1.1 Approximating $R^{(\infty)}$ by using $D^{(\infty)}$

The stationary r was introduced on Page 49, denoted by $R^{(\infty)}$. The relation of the distance r to the quantity D on the sphere model was established on Page 37. The $D^{(\infty)}$ values for different ES algorithms are given on Page 63. Based on the relation between $R^{(\infty)}$ and $D^{(\infty)}$, the $R^{(\infty)}$ values can be approximated by using $D^{(\infty)}$, and using $N-1$ instead of N.

To give an example, the approximate $R^{(\infty)}$ value for the $(1, \lambda)$-ES reads

$$R^{(\infty)}_{1,\lambda} \approx \frac{(N-1)\sigma}{2c_{1,\lambda}} \ . \tag{6.41}$$

The validity of this approximation is dependent on σ, as will be seen in Section 6.4.

The stationary value $R^{(\infty)}$ has an important role in the analysis of ES algorithms on ridge functions. The term with r dominates the fitness value for ridge functions with $\alpha > 1$ if $\sigma \gg 1$ (cf. $F_R(\mathbf{x})$ definition in Equation (3.13)). Therefore, one expects in this $N-1$ dimensional subspace a progress behavior similar to the one on the sphere model. In Point 6.4.1.4, this assumption will be verified for the $(1, \lambda)$-ES, by comparing the actual analytical formula with the first level estimates from the sphere model theory for large values of the mutation strength: Equation (6.166) asymptotically ($\sigma \to \infty$) becomes equivalent to (6.41). A similar behavior is observed in Section 6.4 for the respective $R^{(\infty)}$ values of other ES algorithms. For small σ values, the actual stationary distance is even larger, this means that the isofitness line can be approximated even better by the hyperplane since the local curvature decreases further for larger r. The effect of d on $R^{(\infty)}$ can be seen in the analytically derived formulae in Section 6.4, and therefore it will not be discussed here in detail.

6.3.1.2 Local approximation by hyperplane

The local property of the isofitness surface can be approximated by a hyperplane (see e.g. Figure 3.5). A typical plot is shown in Figure 6.1. In this figure, the stationary case is depicted for the parabolic ridge with $d = 0.01$, $N = 100$, and $\sigma = 8$. The expected length $\|\mathbf{z}\|$ of a mutation vector can be calculated by using the Central Limit Theorem [Roh76, p. 282] since all N variables are mutated normally (cf. Equation (6.174)). Hence, the length of the mutation vector is roughly the square root of the sum of variances for all variables. One obtains $\|\mathbf{z}\| \approx \sigma\sqrt{N} = 80$, where the relative error in this result vanishes for $N \to \infty$.

The relevant part of Figure 6.1 is focused in Figure 6.2. Another mutation vector $\mathbf{z}$ is shown here. The difference between the isofitness line and the hyperplane is exaggerated, and the vector $\mathbf{z}$ is shown in a fraction of its original size. In scope of this approximation, the isofitness line of the hyperplane is considered in the analysis, instead of the isofitness curve of the parabolic ridge. This is admissible because they locally do not differ much. Consequently, one expects for the parabolic ridge a progress rate equivalent to the one of the hyperplane, however, in the direction of the gradient $\mathbf{a}$. This expected progress is indicated as $\boldsymbol{\varphi}$.

Since the progress rate is measured in the direction of the ridge axis (indicated as $\mathbf{e}_0$ in the figure), one has to consider only the $\mathbf{e}_0$ component of the vector $\boldsymbol{\varphi}$. As a result, the component orthogonal to the ridge axis cannot be considered as progress. These $N-1$ directions are represented in the figure by the unit vector $\mathbf{e}_r$. The next step is the formal calculation of these two components of the vector $\boldsymbol{\varphi}$.

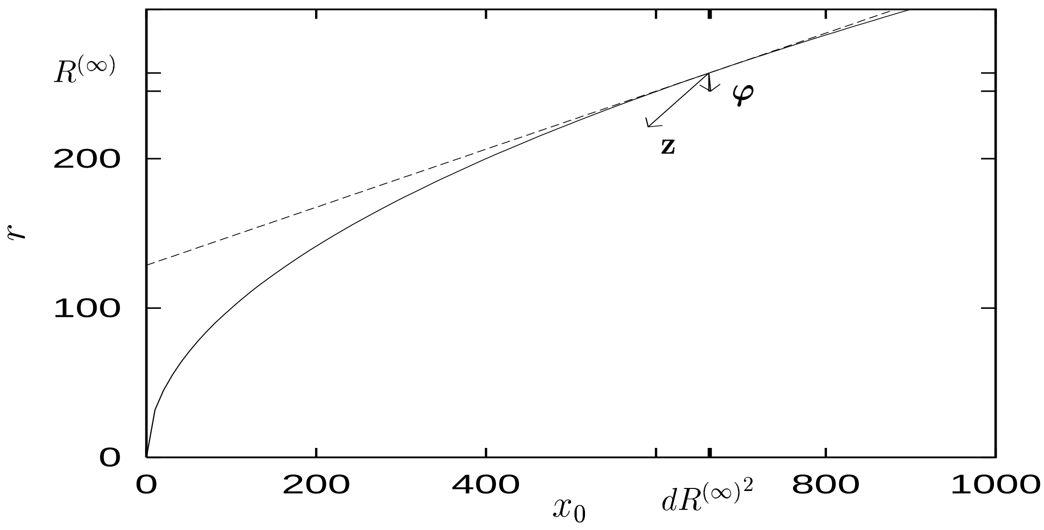

Figure 6.1: The search space plotted r versus x_0 for the parabolic ridge ($d = 0.01$, $N = 100$). The isofitness line and the approximating hyperplane are shown. The progress vector (the almost vertical one) and an arbitrarily chosen mutation vector (the longer one) for the stationary case ($r \approx R^{(\infty)}$, $\sigma = 8$) are depicted. The value $R^{(\infty)} \approx 257.4$ is obtained using (6.41), which is used in the figure. Equation (6.166) yields 259 for these values. The length of the mutation vector is $\|\mathbf{z}\| \approx \sigma\sqrt{N} = 80$.

6.3.1.3 The $(1, \lambda)$-ES case

In Point 6.3.1.2, the fitness landscape of the parabolic ridge has been locally approximated by an hyperplane for the stationary case. This model can also be applied to ridge functions with $\alpha > 2$ since the curvature of the isofitness line for the stationary distance $R^{(\infty)}$ is even smaller for these functions (cf. Figure 3.7 for $\alpha = 10$). Such small curvature values yield a better local approximation by a hyperplane. Additionally, this model is also applicable for the stationary case of ridge functions with $\alpha < 2$, since the curvature of these functions is less than the one of the parabolic ridge for the same r. Furthermore, their observed $R^{(\infty)}$ value can be larger than the one for the parabolic ridge. The value of $R^{(\infty)}$ for both cases will be investigated by experiments in Subsection 7.1.3. The stationary φ value for a given α can be obtained as first order approximation by inserting the empirical $R^{(\infty)}$ value into the φ formulae obtained in this subsection using the local model.

The progress rate φ for the $(1, \lambda)$-ES on ridge functions can be calculated approximately for the stationary case using the local model introduced in Figure 6.2. The progress rate φ of the hyperplane (Equation (5.22)) is written in vector form as

$$\boldsymbol{\varphi} = \frac{\mathbf{a}}{\|\mathbf{a}\|}\varphi = \frac{\mathbf{a}}{\|\mathbf{a}\|}c_{1,\lambda}\sigma \quad . \tag{6.42}$$

By using $\mathbf{e}_0$ and $\mathbf{e}_r$, and the definition of r in (3.12), the gradient vector $\mathbf{a}$ (Equation (6.1))

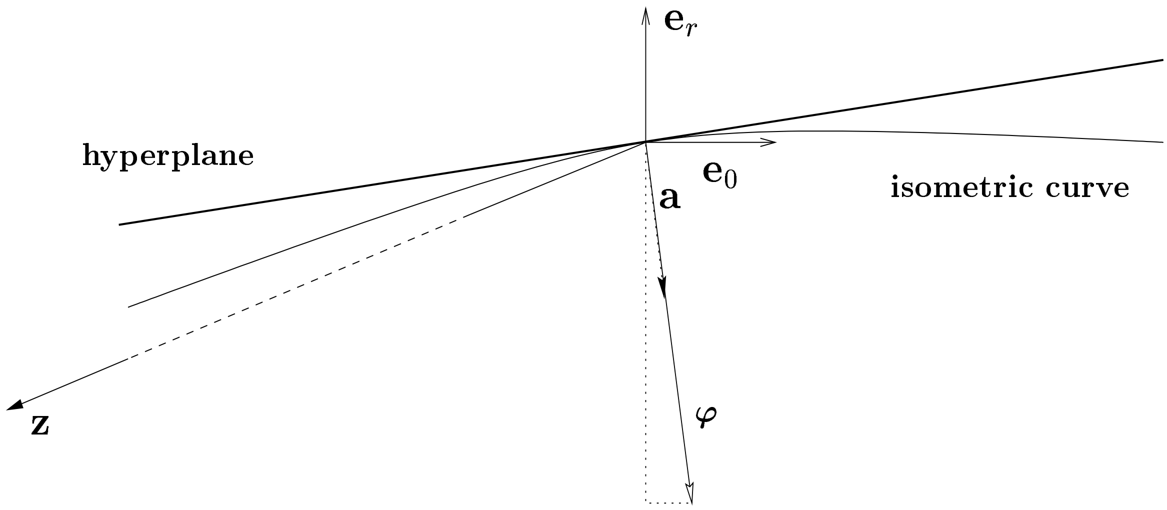

Figure 6.2: The local approximation for the stationary case. This figure magnifies the relevant parts of Figure 6.1, however uses a different mutation vector $\mathbf{z}$, shown as a fraction of its actual length. The unit vectors $\mathbf{e}_0$ and $\mathbf{e}_r$, the progress vector $\boldsymbol{\varphi}$ for the hyperplane, and the gradient vector $\mathbf{a}$ are shown. The progress rate φ of the parabolic ridge is the component of $\boldsymbol{\varphi}$ in $\mathbf{e}_0$ direction.

can be rewritten as

$$\mathbf{a} = \begin{pmatrix} \frac{\partial F}{\partial x_0} \\ \\ \frac{\partial F}{\partial r} \end{pmatrix} = \begin{pmatrix} 1 \\ -d\alpha r^{\alpha-1} \end{pmatrix} = \mathbf{e}_0 - d\alpha r^{\alpha-1}\mathbf{e}_r \ . \tag{6.43}$$

The component of $\boldsymbol{\varphi}$ in $\mathbf{e}_0$ direction gives the stationary progress rate value of the $(1,\lambda)$-ES on the ridge functions

$$\varphi \approx \mathbf{e}_0^{\mathrm{T}} \cdot \boldsymbol{\varphi} = \frac{c_{1,\lambda}\sigma}{||\mathbf{a}||}\mathbf{e}_0^{\mathrm{T}} \cdot \mathbf{a} = \frac{c_{1,\lambda}\sigma}{||\mathbf{a}||} \ , \tag{6.44}$$

and therefore

$$\boxed{\varphi \approx \frac{c_{1,\lambda}\sigma}{\sqrt{1+(d\alpha r^{\alpha-1})^2}} \ .} \tag{6.45}$$

This result was obtained by assuming that the isometric hypersurface can locally be approximated by a hyperplane. For $\alpha = 0$, (6.45) can be investigated further. It becomes an exact equation (see Equation (5.22)).

Please note that this static progress rate formula is also applicable for any arbitrary r value (see Subsection 7.2.10). The static progress rate formula of the $(1,\lambda)$-ES will be derived in Subsection 6.3.2 for a given r value. This static result (6.84) can be used for the stationary case, using the $R^{(\infty)}$ value which will be derived in Subsection 6.4.1. Actually, the theoretical $R^{(\infty)}$ value can also be inserted into (6.45). Both formulae will be compared with the simulation results in Section 7.2.

In the following, the asymptotic ($\sigma \to \infty$) properties of (6.45) will be investigated by assuming that $R^{(\infty)}$ can be approximated by $D^{(\infty)}$ in (6.41). The reader is referred to Point 6.3.1.1 for explanations on the applicability of this approximation. The parabolic ridge case ($\alpha = 2$) is considered first. After substituting (6.41) for r, the steady state progress rate formula reads

$$\varphi \approx \frac{c_{1,\lambda}\sigma}{\sqrt{1+(2dr(\sigma))^2}} = \frac{c_{1,\lambda}\sigma}{1+\left(d\frac{(N-1)\sigma}{c_{1,\lambda}}\right)^2} \ . \tag{6.46}$$

Taking the limit one obtains the asymptotic progress rate φ for the parabolic ridge

$$\lim_{\sigma\to\infty} \varphi \approx \frac{c_{1,\lambda}^2}{d(N-1)} \ . \tag{6.47}$$

This result for φ will be shown to be the maximum possible static progress rate $\hat{\varphi}$ for the $(1,\lambda)$-ES. The appropriateness of this asymptotic equality will be shown in the next subsection (see Equation (6.90)). By introducing a normalization, this result can be expressed independent of the parameters d and N. Furthermore, a similar normalization for the mutations strength σ will make the stationary progress rate formula simpler:

$$\boxed{\sigma^* := d(N-1)\sigma, \qquad \varphi^* := d(N-1)\varphi} \tag{6.48}$$

This normalization is of different nature than the one for the sphere model (Equation (5.25)). It will be used to simplify formulae and to generalize simulation results (cf. Subsection 4.4 for the benefits of normalization). As an example, the limit of the normalized progress rate φ^* will be computed for the parabolic ridge case. Applying (6.48) to (6.46), one gets

$$\lim_{\sigma^*\to\infty} \varphi^* \approx \lim_{\sigma^*\to\infty} \frac{c_{1,\lambda}\sigma^*}{\sqrt{1+(\frac{\sigma^*}{c_{1,\lambda}})^2}} = c_{1,\lambda}^2 \ . \tag{6.49}$$

For the general case of ridge functions, the limit for (6.45) becomes using (6.41)

$$\begin{aligned} \lim_{\sigma\to\infty} \varphi \quad &\approx \quad \lim_{\sigma\to\infty} \frac{c_{1,\lambda}\sigma}{\sqrt{1+(\alpha d r^{\alpha-1})^2}} \\ &\approx \quad \lim_{\sigma\to\infty} \frac{c_{1,\lambda}\sigma}{\sqrt{1+\left(\alpha d\left[\frac{(N-1)\sigma}{2c_{1,\lambda}}\right]^{\alpha-1}\right)^2}} = \lim_{\sigma\to\infty} \frac{c_{1,\lambda}\sigma}{\alpha d}\left(\frac{2c_{1,\lambda}}{(N-1)\sigma}\right)^{\alpha-1} \ . \end{aligned} \tag{6.50}$$

In this partial result, a normalization of σ similar to (6.48) would again give an equation independent of d and N. This normalization must be nonlinear in d. One obtains

$$\sigma^* := d^{\frac{1}{\alpha-1}}(N-1)\sigma, \qquad \varphi^* := d^{\frac{1}{\alpha-1}}(N-1)\varphi \ . \tag{6.51}$$

In the simulations chapter, this normalization will be used to compare the progress results obtained for different ridge functions. Such a comparison cannot be made theoretically, since a general $R^{(\infty)}$ formula has not been derived yet.

Unfortunately, this general normalization cannot be applied to the sharp ridge case ($\alpha = 1$). However, (6.45) becomes for this case so simple that no normalization is needed

$$\varphi|_{\alpha=1} \approx \frac{c_{1,\lambda}\sigma}{\sqrt{1+d^2}} \quad . \tag{6.52}$$

This result will be investigated by experiments in Subsection 7.2.5.

Discussion. The results obtained for the $(1,\lambda)$-ES on ridge functions using the local model in Figure 6.2 for the stationary case ($r \approx R^{(\infty)}$) will be discussed.

The progress rate of the parabolic ridge is expected to go to a nonzero limit as the mutation strength goes to infinity (cf. (6.47) and (6.49)). This is a new result as compared to the limits observed for the hyperplane and sphere model: For the hyperplane case, the limit $\lim_{\sigma\to\infty}\varphi$ goes to infinity (see Equation (5.22)), and for the sphere model, it goes to $-\infty$ for constant R and N (see Equation (5.26) and the normalization in (5.25)). Therefore, one observes a different convergence behavior for the parabolic ridge.

The same limit can also be investigated for cases other than $\alpha = 2$, using the asymptotic approximation for $R^{(\infty)}$ in (6.41). For $\alpha < 2$, the numerator of (6.45) is of higher order in σ than the denominator, as can be seen in (6.50). As a result, the limit will be infinite. The hyperplane case ($\alpha = 0$) is included in this case. As a by-product, the progress rate for $\alpha < 0$ becomes equivalent to the one of the hyperplane. In this case, the effect of r becomes negligible in the denominator: Since it has a negative exponent, the second term in (6.45) can be neglected as compared to the first one.

Since progress is measured in the direction specified by the ridge axis, the ES algorithm cannot get a higher progress rate value for $\alpha < 0$ than it attains for the hyperplane. The progress coefficient defines the progress limit for a given direction and unit mutation strength. Therefore, the case $\alpha < 0$ (concave functions) is not interesting for the investigation of the progress rate.

On the other case, $\alpha > 2$, the dominator of (6.45) is of higher order in σ (see also Equation (6.50)); therefore, the limit $\lim_{\sigma\to\infty}\varphi$ is zero. Since φ is zero for $\sigma = 0$ and for $\sigma \to \infty$, one expects a maximum for a value of σ in-between. The values for the optimum progress rate $\hat{\varphi}$ and optimum mutation rate $\hat{\sigma}$ (or $\hat{\varphi^*}$ and $\hat{\sigma^*}$) can be determined numerically. For $\alpha = 2$, one obtains $\hat{\sigma} = \hat{\sigma^*} = \infty$ and therefore the $\hat{\varphi}$ value in (6.47) and the $\hat{\varphi^*}$ value in (6.49), respectively. The proof of this fact will follow in Subsection 6.3.2.

From the φ formula (6.45), one can infer that the (stationary) progress rate for any ridge function must be nonnegative, independent of the values of σ, d, N, and r. It becomes also clear that the *static* value of the progress rate is maximal for $r = 0$ (in other words, on the ridge axis). For $r \to \infty$, the progress rate approaches *zero* for the $\alpha > 1$ case. The cases $\alpha = \{1, 2, 8\}$ will be investigated in Subsection 7.2.10.

Other ES algorithms. Up to now, the $(1,\lambda)$-ES algorithm was considered. Actually, the local model in Figure 6.2 can be applied to other ES algorithms. As a result, one can obtain approximate formulae for the respective progress rates. Firstly, Equation (6.45) shall be adapted to other algorithms. This will be done basically by using the appropriate progress coefficient in the numerator of (6.45). As a result, the approximate stationary formulae for the general ridge function can simply be obtained.

In the following, only the $\alpha = 2$ case is shown in order to reduce the number of formulae to be considered, and since the asymptotic ($\sigma \to \infty$) limit of the stationary progress rate is of utmost interest for different algorithms. This limit will be determined for each algorithm considered. For the $(1,\lambda)$-ES, this limit was obtained as ∞ for $\alpha < 2$ and as 0 for $\alpha > 2$, respectively. These two limits will also be valid for other ES algorithms.

The derivation of the progress rate formulae for other ES algorithms are based on the same two assumptions made for the $(1,\lambda)$-ES: The local isofitness curve is approximated by a hyperplane. Thereafter, the progress rate for this hyperplane is decomposed using the evolutionary progress principle. In the second step, the component giving the progress rate of the parabolic ridge is obtained using the gradient vector.

6.3.1.4 The $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES

The progress rate of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES on the hyperplane was given in (5.24). In this algorithm, the mutations are generated from the centroid of μ parents in the search space (see Subsection 2.4.1 for the algorithm). Therefore, the distance r corresponds to the distance of this centroid to the ridge axis. Consequently, the formula for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES reads

$$\varphi \approx \frac{c_{\mu/\mu,\lambda}\sigma}{\sqrt{1+(2dr)^2}} \; . \tag{6.53}$$

The stationary value of r is approximated by using the $D^{(\infty)}$ value in (5.32). Similar to the considerations in Point 6.3.1.1, and using (6.48) for the normalization, the approximate stationary value $R^{(\infty)}$ is inserted in (6.53). The asymptotic limits read

$$\lim_{\sigma\to\infty} \varphi \approx \lim_{\sigma\to\infty} \frac{c_{\mu/\mu,\lambda}\sigma}{\sqrt{1+\left(2d\frac{(N-1)\sigma}{2\mu c_{\mu/\mu,\lambda}}\right)^2}} = \frac{\mu c^2_{\mu/\mu,\lambda}}{d(N-1)}, \qquad \lim_{\sigma^*\to\infty} \varphi^* \approx \mu c^2_{\mu/\mu,\lambda} \; . \tag{6.54}$$

These asymptotic limits will be calculated formally using the asymptotically ($N \to \infty$) exact formulae for φ and $R^{(\infty)}$ of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES in Subsection 6.3.3. The values obtained accord the ones in (6.54).

6.3.1.5 The $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES

Up to now, the full analysis of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES is an open problem in the ES theory. In [Rec94, p. 150], it is asserted that the effects of dominant and intermediate recombination are the same if the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES operates at a mutation strength $\sqrt{\mu}$ times larger than the σ value for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES. An approximate theoretical approach for $N \to \infty$ to explain this observation can be found in [Bey95a], [Bey96c, p. 227-235] which uses a *surrogate mutation* model. The idea of surrogate mutations consists of substituting the effect of mutation and recombination operators with a mutation operator alone. The resulting mutations are applied to the centroid of the population. This centroid is virtual since the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES does not compose the centroid explicitly. The first step is finding the strength σ_s of the surrogate mutations. The asymptotic ($N \to \infty$) result is obtained by Beyer as $\sigma_s = \sqrt{\mu}\sigma$ for the stationary distribution of the population. The statistical estimate for the strength of the surrogate mutations verifies the relation proposed by Rechenberg. This relation has an hypothetical character, and it is not formally proven yet. The necessary conditions for its correctness are unknown. The progress rate formula for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES is obtained by substituting the mutation strength by $\sqrt{\mu}\sigma$ in the progress rate formula in (6.53) for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES. The resulting formula reads

$$\varphi \approx \frac{\sqrt{\mu} c_{\mu/\mu,\lambda} \sigma}{\sqrt{1 + (2dr)^2}} \ . \tag{6.55}$$

The asymptotic ($\sigma \to \infty$) value for the stationary case is obtained by approximating $R^{(\infty)}$ by $D^{(\infty)}$. The same consideration has also been used for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES in Point 6.3.1.4, it was explained in detail in Point 6.3.1.1. The respective $D^{(\infty)}$ formula can be found in (5.33). Therefore, using the normalization in (6.48), one gets

$$\lim_{\sigma \to \infty} \varphi \approx \lim_{\sigma \to \infty} \frac{\sqrt{\mu} c_{\mu/\mu,\lambda} \sigma}{\sqrt{1 + \left(2d \frac{(N-1)\sigma}{2\sqrt{\mu} c_{\mu/\mu,\lambda}}\right)^2}} = \frac{\mu c^2_{\mu/\mu,\lambda}}{d(N-1)}, \qquad \lim_{\sigma^* \to \infty} \varphi^* \approx \mu c^2_{\mu/\mu,\lambda} \ . \tag{6.56}$$

Please note that the formulae in (6.55) and (6.56) are valid for the parabolic ridge if the ridge axis is diagonally placed in the N dimensional search space (see Equation (3.17) for the definition of the rotated ridge function). If the unit vector $\mathbf{v}$ shows a different direction, a lower progress rate is observed in simulations. Even for the diagonal case, the derivation has an hypothetical character.

For the explanation of this observation, one has to look closer to the local model (Figure 6.2 on Page 88). This model was used to derive the progress rate formulae for the ridge functions. It is based on locally approximating the isofitness curve by a hyperplane. Since the progress direction is given by the unit vector $\mathbf{v}$, the component of the progress vector of the hyperplane in the $\mathbf{v}$ direction gave us the progress rate of the ES algorithm.

The progress rates of all ES algorithms do not depend on how the coordinate axes are placed, except the algorithm which uses dominant recombination. This statement is valid for isotropic mutations, and for the cases both with or without intermediate recombination. However, the progress performance of ES algorithms *with* dominant recombination is affected by rotations. The investigation of the general case for the diagonal $\mathbf{v}$ and a formal proof that (6.55) gives the maximum progress rate for any $\mathbf{v}$ could not be done in scope of this work. Some simulation results obtained using the $(\mu/\mu_D, \lambda)$-ES on ridge functions can be found in Subsection 7.2.4 and Subsection 7.2.7. In the following, the progress rate of the $(\mu/\mu_D, \lambda)$-ES will be considered for the hyperplane case ($\alpha = 0$).

6.3.1.6 The $(\mu/\mu_D, \lambda)$-ES on the hyperplane

The progress rate φ will be considered for two cases using the same σ. Although unproven, these two cases seem to yield the two extreme values for the progress rate: If the unit vector $\mathbf{v}$ indicating the progress direction is aligned with the unit vector for a variable, φ is minimized; *however*, if it is diagonally placed with respect to the unit vectors for variables, it is maximized. These two cases can be formalized as follows for the general rotated hyperplane in (3.21):

$$1. \qquad \mathbf{v} := (v_0, v_1, \ldots, v_{N-1})^{\mathrm{T}}, \qquad v_i = \pm 1 \text{ and } \forall k \neq i: \; v_k = 0 \tag{6.57}$$

$$2. \qquad \mathbf{v} := \frac{1}{\sqrt{N}}(v_0, v_1, \ldots, v_{N-1})^{\mathrm{T}}, \qquad v_i = \pm 1 \tag{6.58}$$

In the former case, there are $2N$ different choices for $\mathbf{v}$. The progress axis is aligned with the unit vector of a variable. As a result, other $N-1$ variables do not appear in the fitness function (see e.g. Equation (3.14)). Consequently, they are selection-invariant. The selection is done by using the value of the single variable x_i with nonzero v_i only.

It is interesting to investigate the operation of the $(\mu/\mu_D, \lambda)$-ES algorithm on this single variable: As can be seen in Algorithm 5 on Page 11, the recombination operator is applied before the mutation operator. It produces a temporary state, and the mutation operator is applied later at this state. In dominant recombination (see Subsection 2.4.2), each component of this temporary state is selected randomly in the parental pool. Since $\mathbf{v}$ is aligned, $N-1$ of these components are selection-invariant. The variable which is used in the fitness evaluation is selected from the pool of μ parents, and the mutation is applied thereafter. However, this is exactly the way how the (μ, λ)-ES operates. As a consequence, the $(\mu/\mu_D, \lambda)$-ES gives exactly the same progress rate as the (μ, λ)-ES, i.e. $\varphi = c_{\mu,\lambda}\sigma$ as given in (5.23).

The latter case shown in (6.58) describes the vector $\mathbf{v}$ diagonal in the search space spanned by the unit vectors for variables. For the realization, one has 2^N different choices for $\mathbf{v}$. At any of these choices, all variables will have the same effect on the fitness value (see Equation 3.21). The population distribution of the parental generation is necessary for the exact calculation of the progress rate φ. Alternatively, one can model the effect of mutation and recombination operators by a surrogate mutation. An overview to this model was given in Point 6.3.1.5. The progress rate for dominant recombination can be obtained by substituting σ by $\sqrt{\mu}\sigma$ in the progress rate formula for the $(\mu/\mu_I, \lambda)$-ES. In summary, one obtains the progress rate of the $(\mu/\mu_D, \lambda)$-ES on the hyperplane for the two cases in (6.57) and (6.58) as follows

$$\boxed{\varphi = \begin{cases} c_{\mu,\lambda}\sigma & \text{if } \mathbf{v} \text{ is aligned} \\ \sqrt{\mu}c_{\mu/\mu,\lambda}\sigma & \text{if } \mathbf{v} \text{ is diagonal.} \end{cases}} \tag{6.59}$$

This result will be compared with simulation results in Subsection 7.2.7. Please note that the result for the aligned $\mathbf{v}$ is valid for any N. For the diagonal case, it is obtained asymptotically for $N \to \infty$.

6.3.1.7 The (μ, λ)-ES

The observations on the local model will be finished with the (μ, λ)-ES case. The performance of the (μ, λ)-ES on ridge functions is estimated using the same scheme already used above for other algorithms. Considering the progress rate of this algorithm on the hyperplane (Equation (5.23)), the local model in Figure 6.2 on Page 88, and the progress rate formula in (6.45), one obtains the progress rate of the (μ, λ)-ES as

$$\varphi \approx \frac{c_{\mu,\lambda}\sigma}{\sqrt{1+(2dr)^2}} \quad . \tag{6.60}$$

The approximation of $R^{(\infty)}$ by $D^{(\infty)}$ was explained in Point 6.3.1.1. After inserting the stationary value $D^{(\infty)}$ (Equation (5.31)) in (6.60) for r, the asymptotic limit for the stationary progress rate can be computed (consider (6.48) for the normalization)

$$\lim_{\sigma\to\infty} \varphi \approx \lim_{\sigma\to\infty} \frac{c_{\mu,\lambda}\sigma}{\sqrt{1+\left(2d\frac{(N-1)\sigma}{2c_{\mu,\lambda}}\right)^2}} = \frac{c_{\mu,\lambda}^2}{d(N-1)}, \qquad \lim_{\sigma^*\to\infty} \varphi^* \approx c_{\mu,\lambda}^2 \quad . \tag{6.61}$$

6.3.1.8 Summary

In this subsection, the progress rate of general ridge functions has been investigated using a simple local model. As a result, a progress rate formula (Equation (6.45)) has been obtained for the stationary case. This formula can be adapted for other ES algorithms, as it is done subsequently for the parabolic ridge case. For other ridge functions, this extension is also possible (cf. the two paragraphs ending Point 6.3.1.3).

The progress rate of ES algorithms strongly depends on the value α. The asymptotic limit of φ is observed for $\sigma \to \infty$ on the parabolic ridge. It is *finite* for the ES algorithms investigated, although the same limit gives $-\infty$ for the sphere model and ∞ for the hyperplane. One observes that this limit for the (μ, λ)-ES is smaller than the one for the $(1, \lambda)$-ES. Additionally, it is equal for the $(\mu/\mu_\mathrm{I}, \lambda)$-ES and for the $(\mu/\mu_\mathrm{D}, \lambda)$-ES if the unit vector $\mathbf{v}$ is chosen to be diagonal for the $(\mu/\mu_\mathrm{D}, \lambda)$-ES. The value obtained for these two latter cases is larger than the $(1, \lambda)$-ES case depending on the μ value used. In the next two subsections, it will be shown that this asymptotic limit ($\sigma \to \infty$) gives the maximum stationary progress rate $\hat{\varphi}$ for the parabolic ridge. In Subsection 6.3.6, the progress efficiency η will be used to compare the progress rates of different ES algorithms.

The stationary progress rate formulae for the $(1, \lambda)$-ES on the sharp ridge is also obtained as a by-product of the analysis (Equation (6.52)). Similarly, the φ formulae for other ES algorithms can be obtained by using corresponding values for $R^{(\infty)}$ and the progress coefficient, as it is done here for the parabolic ridge case.

The $(\mu/\mu_D, \lambda)$-ES algorithm has been discussed for the hyperplane case in Point 6.3.1.6. Two different progress rate formulae are obtained depending on the direction of the vector $\mathbf{v}$. The performance of the $(\mu/\mu_D, \lambda)$-ES depends on the $\mathbf{v}$ value also for other ridge functions, as will be seen in Subsection 7.2.4 and Subsection 7.2.7.

The analytical derivation of the progress rate formulae for the $(1, \lambda)$-ES and for the $(\mu/\mu_I, \lambda)$-ES on the general ridge function follows in the next two subsections, respectively. The obtained results are asymptotically ($N \to \infty$) exact for $N \gg \lambda$. The analytical derivation of the two respective $R^{(\infty)}$ values will be carried out in Section 6.4. The formulae asserted for the $(\mu/\mu_D, \lambda)$-ES and for the (μ, λ)-ES can also be found there.

6.3.2 The $(1, \lambda)$-ES

In this subsection, the method described in Subsection 5.3.7 will be used to derive the progress rate φ for the $(1, \lambda)$-ES on the general ridge function $F_{RR}(\mathbf{x})$ (Equation 3.17). The principal idea of induced order statistics [Bey93, Bey96c] is used for the analytical calculation of the progress component of the mutations that generated the selected individuals; i.e. the expected progress in the search space obtained by fitness-based selection. Only the effect of mutation and selection is considered in this subsection. The analysis for the $(\mu/\mu_I, \lambda)$-ES will follow in Subsection 6.3.3.

The progress rate of the $(1, \lambda)$-ES can be expressed as an expected value integral

$$\varphi = \mathrm{E}\{z\} = \int_{-\infty}^{\infty} z\, p_{1,\lambda}(z)\, \mathrm{d}z \ . \tag{6.62}$$

The random variable z describes the progress toward the optimum in the search space. In other words, it is the component of the mutation generating the best descendant (with respect to the fitness) in the progress direction. This direction is given by the unit vector $\mathbf{v}$. The integral is taken over all possible values of z. Since the functioning mechanism of the $(1, \lambda)$-ES with isotropic mutations is *not* affected by rotations, it will be assumed that the progress direction coincides with a coordinate axis (cf. Equation (3.11)), and the progress is measured using the values for this variable.

The probability density of z is described by $p_{1,\lambda}(z)$, where the ES algorithm is indicated in the subscript. This density is to be specified next. All λ descendants are generated using isotropic mutations (cf. Equations (2.1)–(2.3)). Therefore, the offspring are normally-distributed in any arbitrary direction, indicated as $p_z(z) \sim \mathcal{N}(0, \sigma^2)$ (cf. Equation (5.4)). This is also true for the descendant with the best fitness value among all λ offspring, i.e. the best individual is also generated by such a mutation. The component of this mutation in the progress direction is described by the variable z in (6.62). Per definition, the best individual has a fitness value better than all remaining $\lambda - 1$ descendants generated. This condition can be expressed by the acceptance probability $P_{a\,1,\lambda}(z)$. Anyone of λ descendants can principally be the best one; therefore, the product $p_z(z)P_{a\,1,\lambda}(z)$ must be multiplied by λ. As a result, the pdf $p_{1,\lambda}(z)$ reads

$$p_{1,\lambda}(z) = \lambda \cdot p_z(z)\, P_{a\,1,\lambda}(z) \ . \tag{6.63}$$

The cdf $P_{a\,1,\lambda}(z)$ remains to be determined. It describes the acceptance probability for a given z value; i.e. the probability that the fitness of an individual with this z value is better than all other $\lambda - 1$ fitness values. Equivalently, $P_{a\,1,\lambda}(z)$ can be defined using local quality function (LQF) values, since the parent of all descendants is the same. The LQF was defined in (4.3). In the following, the LQF value for a given progress component z will be denoted by $Q_{|z}$, which can take any real value. The pdf of $Q_{|z}$ can be written as $p(Q_{|z}|z)$. The probability that a descendant has an LQF value less (worse) than $Q_{|z}$ will be denoted by $P_1(Q_{|z})$; consequently, this probability will be $[P_1(Q_{|z})]^{\lambda-1}$ for $\lambda - 1$ descendants. The resulting integral for the cdf $P_{a\,1,\lambda}(z)$ reads

$$P_{a\,1,\lambda}(z) = \int_{-\infty}^{\infty} p(Q_{|z}|z)\,[P_1(Q_{|z})]^{\lambda-1} \mathrm{d}Q_{|z} \ . \tag{6.64}$$

The $Q_{|z}$ value for a given z can practically have all possible values for the LQF; the interval for these possible values is therefore considered as $(-\infty, \infty)$, as for the range of z in (6.62). This range is appropriate in both cases for ridge functions: The optimum is at infinity, therefore the z range is appropriate. The interval for $Q_{|z}$ was formally discussed for Equation (4.5), and it is bounded depending on the worst and best possible fitness values. Since both of these values are unbounded for ridge functions, the $Q_{|z}$ value can take all possible real values.

The progress rate φ can be expressed after combining the partial results in (6.62), (6.63), and (6.64)

$$\varphi = \lambda \int_{-\infty}^{\infty} z\, p_z(z)\, P_{a\,1,\lambda}(z)\, \mathrm{d}z = \lambda \int_{-\infty}^{\infty} z\, p_z(z) \int_{-\infty}^{\infty} p(Q_{|z}|z)\,[P_1(Q_{|z})]^{\lambda-1}\, \mathrm{d}Q_{|z}\, \mathrm{d}z \ . \tag{6.65}$$

One can observe an important point in (6.65): The complicated expression $[P_1(Q_{|z})]^{\lambda-1}$ does not directly depend on z. Therefore, one can exchange the integration order, and substitute $Q := Q_{|z}$; resulting

$$\varphi = \lambda \int_{-\infty}^{\infty} [P_1(Q)]^{\lambda-1} \left[\int_{-\infty}^{\infty} z\, p_z(z)\, p(Q|z)\, \mathrm{d}z\right] \mathrm{d}Q \ . \tag{6.66}$$

This equation is the starting point of the progress analysis for the $(1,\lambda)$-ES, and was already given in (5.50).

The functions $P_1(Q)$ and $p(Q|z)$ are fitness-dependent; therefore, they must be determined anew for each fitness function of interest. A first order approximation to $P_1(Q)$ is given in (5.51), $P_1(Q) \approx \Phi[(Q - M_Q)/S_Q]$. The M_Q and S_Q values can be calculated for any function, as shown in (5.40). However, this approximation may not be accurate for all fitness functions. This case was already mentioned in Subsection 5.3.3. The cdf $P_z(z)$ in (5.17) is nothing but the normalized form of the distribution $P_1(Q)$. If the simplest approximation in (5.51) is not accurate enough, one has to use the additive terms given. As a result, (6.66) becomes much harder to integrate.

The density $p(Q|z)$ is obtained using the LQF by keeping z (the component of the mutation vector in progress direction) as a conditional variable. The mean value and standard deviation of the LQF are easy to calculate. The density $p(Q|z)$ is approximated by a normal distribution conditional to z. The mutations are normally distributed in the search space; however, the fitness values are in general not normally distributed in the fitness space. Since $p(Q|z)$ is defined in the fitness space, this normal approximation is not valid for any arbitrary function. As in any approximation, a computation error is caused by the normal approximation of $p(Q|z)$. The theoretical formulae obtained for φ are therefore compared with the simulation results in Section 7.2. The results obtained are quite satisfactory.

6.3.2.1 The calculation of $P_1(Q)$

At the end of Subsection 5.3.7, the approximation $P_1(Q) \approx \Phi[(Q - M_Q)/S_Q]$ was suggested in (5.51). The error made by this approximation depends on the fitness function analyzed. In the following, the cdf $P_1(Q)$ is derived using the density functions $p(Q_{|z}|z)$ and $p_z(z)$. The $P_1(Q)$ is obtained from the definition of conditional probability functions

$$P_1(Q) := \int_{-\infty}^{Q} \int_{-\infty}^{\infty} p(Q_{|z}|z)\, p_z(z)\, \mathrm{d}z\, \mathrm{d}Q_{|z} \quad . \tag{6.67}$$

The inner integral is taken for all possible values of z, it gives the expected value of the density $p(Q_{|z}|z)$ for all values of z. The outer integral gives the probability that this expected value is below a given value Q, i.e. the cdf for the values of $Q_{|z}$ less than Q. The probability density $p_z(z)$ was already used in (6.63). After the inner integral is taken over z, the outer integral and the final result is no more dependent on z. The conditional pdf $p(Q_{|z}|z)$ must be determined next and then the integration of (6.67) can be carried out.

The density $p(Q_{|z}|z)$ is approximated by the normal distribution. Therefore, the mean value $M_{Q|z}$ and the standard deviation $S_{Q|z}$ of this conditional density function are to be calculated. The subscript $_{Q|z}$ (instead of $_Q$) indicates that this LQF is conditional to z.

The values of $M_{Q|z}$ and $S_{Q|z}$ are computed next using the conditional LQF; i.e. conditional to the random variable z which presents the mutations in x_0 direction. This LQF is determined for the ridge function (3.11) according to the definition in (4.3). One obtains

$$\begin{aligned}
F_R(\mathbf{x}^{(g)}) &= x_0^{(g)} - d\left[\sum_{i=1}^{N-1} x_i^{(g)2}\right]^{\frac{\alpha}{2}} && (6.68)\\
F_R(\mathbf{x}^{(g+1)}) &= x_0^{(g)} + z_0^{(g)} - d\left[\sum_{i=1}^{N-1} \left(x_i^{(g)} + z_i^{(g)}\right)^2\right]^{\frac{\alpha}{2}} && (6.69)\\
Q(\mathbf{z}) &= F_R(\mathbf{x}^{(g+1)}) - F_R(\mathbf{x}^{(g)})\\
&= z_0^{(g)} - d\left(\left[\sum_{i=1}^{N-1} \left(x_i^{(g)} + z_i^{(g)}\right)^2\right]^{\frac{\alpha}{2}} - \left[\sum_{i=1}^{N-1} x_i^{(g)2}\right]^{\frac{\alpha}{2}}\right) \quad . && (6.70)
\end{aligned}$$

The values of $M_{Q|z}$ and $S_{Q|z}$ are determined conditional to $z_0^{(g)}$ (which is simply denoted as z). The first component of the vector $\mathbf{a}$ (6.1) and the first row and first column of matrix $\mathbf{Q}$ (6.2) are affected by this condition. Therefore, they are renamed as $\mathbf{a}^{'}$ and $\mathbf{Q}^{'}$, respectively. The first component of $\mathbf{a}^{'}$ becomes zero. The matrix $\mathbf{Q}^{'}$ does not differ from $\mathbf{Q}$ since the first row and first column of $\mathbf{Q}$ consist of zeroes anyway. Consequently, the conditional mean value $M_{Q|z}$ is computed as the expected value of the LQF in (6.70). Using (5.40) and (6.17), one obtains

$$M_{Q|z} = z - \sigma^2 \mathrm{Tr}[\mathbf{Q}^{'}] = z - \frac{1}{2} d\alpha\sigma^2 (N + \alpha - 3) r^{\alpha-2} = z + M_Q \ , \tag{6.71}$$

The conditional variable z added is the only difference between M_Q and $M_{Q|z}$, for the unconditional case one has $\mathrm{E}\{z_0^{(g)}\} = \mathrm{E}\{z\} = 0$. The conditional standard deviation $S_{Q|z}$ is obtained using the definition (5.40) and the unconditional S_Q in (6.18). Since the first component of $\mathbf{a}^{'}$ is zero, the first term "1" of the unconditional S_Q does not appear in $S_{Q|z}$. This is the only difference between $S_{Q|z}$ and S_Q. The standard deviation of $z_0^{(g)}$ in (6.70) is zero, since $z_0^{(g)}$ is considered as a constant in the conditional $Q(\mathbf{z})$. One obtains

$$S_{Q|z} = \sigma \sqrt{(d\alpha r^{\alpha-1})^2 + 2\sigma^2 \left[N - 2 + (\alpha-1)^2\right] \left(\frac{d\alpha}{2} r^{\alpha-2}\right)^2} = \sqrt{S_Q^2 - \sigma^2} \ . \tag{6.72}$$

The density $p(Q_{|z}|z)$ is approximated by $\mathcal{N}(M_{Q|z}, S_{Q|z}^2)$. Using (6.67), (6.71), and (6.72), $p(Q_{|z}|z)$ reads

$$p(Q_{|z}|z) \approx \frac{1}{\sqrt{2\pi} S_{Q|z}} \exp\left[-\frac{1}{2} \frac{1}{S_{Q|z}^2} \left(Q_{|z} - M_Q - z\right)^2\right] \ . \tag{6.73}$$

The actual values of $M_{Q|z}$ and $S_{Q|z}$ will not be inserted into the formulae. In the following, their values will be expressed by M_Q and S_Q.

Using the approximation (6.73) for $p(Q_{|z}|z)$, Equation (6.67) can be integrated. In the first step, the integration order is exchanged. Since $p_z(z)$ is independent of $Q_{|z}$, the integration of $p(Q_{|z}|z)$ can be carried out. In the result below, the density $p_z(z)$ is inserted from (5.4)

$$P_1(Q) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} \exp\left[-\frac{1}{2} \frac{z^2}{\sigma^2}\right] \Phi \left[\underbrace{-\frac{\sigma}{S_{Q|z}}}_{a} \frac{z}{\sigma} + \underbrace{\frac{Q - M_Q}{S_{Q|z}}}_{b}\right] \mathrm{d}z \ . \tag{6.74}$$

The substitutions "a" and "b" are introduced for conciseness. After the transformation $t := z/\sigma$, the formula (5.11) can be used to solve the integral

$$P_1(Q) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}t^2} \Phi \left[at + b\right] \mathrm{d}t = \Phi \left(\frac{b}{\sqrt{1 + a^2}}\right) \ . \tag{6.75}$$

The value of the denominator can be calculated as

$$\sqrt{1 + a^2} = \sqrt{1 + \frac{\sigma^2}{S_{Q|z}^2}} = \sqrt{\frac{\sigma^2 + S_{Q|z}^2}{S_{Q|z}^2}} = \frac{S_Q}{S_{Q|z}} \ . \tag{6.76}$$

Therefore, the result for $P_1(Q)$ reads

$$P_1(Q) \approx \Phi\left(\frac{Q - M_Q}{S_Q}\right) \quad . \tag{6.77}$$

Please note that the approximation for $P_1(Q)$ in (6.77) is *exactly* the same as (5.51). The value of M_Q can be found in (6.17), and of S_Q in (6.18), respectively.

In the following, the derivation of the progress rate φ will be carried out for the ridge function class. The distribution $P_1(Q)$ and the density $p(Q|z)$ are not exact for all ridge functions. Therefore, the progress rate formulae obtained using them are valid in scope of the approximation for these probability functions. The resulting formula will mainly be used for the parabolic ridge case, since the LQF approximation in Subsection 5.3.2 is exact for this case. For other ridge functions, however, larger approximation errors are to be expected.

6.3.2.2 The progress rate for ridge functions

The progress rate φ can be obtained for the $(1, \lambda)$-ES by using (6.66) for any fitness function, if the required probability densities are available. In the following, this task will be carried out for the general ridge function. The evaluation has three steps. First, the quantity $P_1(Q)$ must be determined. This has already been done in Point 6.3.2.1 with the result (6.77). It accords to the general first order estimate in (5.51) for any fitness function. The second step is the evaluation of the inner integral in (6.66). For this purpose, the function-specific density $p(Q|z)$ is needed. This density was already used in the derivation of $P_1(Q)$: An approximation to it can be found in (6.73). The inner integral is computed using the integral formula (5.10). Third, the terms will be reordered and the progress coefficient $c_{1,\lambda}$ is substituted from (5.18). After this overview, we can now start with the derivation.

The formulae of $p(Q|z)$ in (6.73) (writing Q instead of $Q_{|z}$) and of $P_1(Q)$ in (6.77) are inserted in the integral (6.66). After using the definition of $p_z(z)$ in (5.4), the result reads

$$\varphi = \frac{\lambda}{2\pi\sigma S_{Q|z}} \int_{-\infty}^{\infty} \left[\Phi\left(\frac{Q - M_Q}{S_Q}\right)\right]^{\lambda-1} \int_{-\infty}^{\infty} z\mathrm{e}^{-\frac{1}{2}\frac{z^2}{\sigma^2}} \exp\left[-\frac{1}{2}\frac{(Q - M_Q - z)^2}{S_{Q|z}^2}\right] \mathrm{d}z\mathrm{d}Q. \tag{6.78}$$

This expression can be simplified by using the substitutions

$$t := \frac{z}{\sigma} \qquad \mathrm{d}t := \frac{\mathrm{d}z}{\sigma} \qquad\qquad s := \frac{Q - M_Q}{S_Q} \qquad \mathrm{d}s := \frac{\mathrm{d}Q}{S_Q} \quad . \tag{6.79}$$

Two further substitutions a and $b(s)$ are introduced

$$\frac{Q - M_Q - z}{S_{Q|z}} = \frac{S_Q \cdot s - z}{S_{Q|z}} = \underbrace{\frac{S_Q}{S_{Q|z}}}_{=:b} s + \underbrace{-\frac{\sigma}{S_{Q|z}}}_{=:a} t = at + b(s) \quad . \tag{6.80}$$

Using (6.79) and (6.80), the expression for the progress rate φ in (6.78) becomes

$$\varphi = \frac{\lambda}{2\pi}\frac{\sigma S_Q}{S_{Q|z}}\int_{-\infty}^{\infty}[\Phi(s)]^{\lambda-1}\int_{-\infty}^{\infty} t\,\mathrm{e}^{-\frac{1}{2}t^2}\exp\left[-\frac{1}{2}(at+b(s))^2\right]\mathrm{d}t\,\mathrm{d}s\ . \tag{6.81}$$

The inner integral is evaluated using the formula (5.10)

$$\begin{aligned}\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} t\,\mathrm{e}^{-\frac{1}{2}t^2}\exp\left[-\frac{1}{2}(at+b(s))^2\right]\mathrm{d}t &= -\frac{ab}{(1+a^2)^{\frac{3}{2}}}\exp\left(-\frac{1}{2}\frac{b^2}{1+a^2}\right)\\ &= \frac{\sigma S_{Q|z}}{S_Q^2}s\,\mathrm{e}^{-\frac{1}{2}s^2}\ .\end{aligned} \tag{6.82}$$

Equation (6.76) is used in the calculation. Thereafter, the result in (6.82) is inserted back in (6.81). After rearranging terms, one obtains

$$\varphi = \frac{\lambda}{\sqrt{2\pi}}\frac{\sigma^2}{S_Q}\int_{-\infty}^{\infty} s\,\mathrm{e}^{-\frac{1}{2}s^2}[\Phi(s)]^{\lambda-1}\mathrm{d}s\ . \tag{6.83}$$

The definition of $c_{1,\lambda}$ (5.18) is easy to identify in (6.83). The value of S_Q can be obtained from (6.18). Thus, one obtains the progress rate of the $(1,\lambda)$-ES for ridge functions

$$\boxed{\varphi = \frac{c_{1,\lambda}\,\sigma^2}{S_Q} = \frac{c_{1,\lambda}\,\sigma}{\sqrt{1+\left(d\alpha r^{\alpha-1}\right)^2+\frac{\sigma^2}{2}\left[N-2+(\alpha-1)^2\right]\left(d\alpha r^{\alpha-2}\right)^2}}\ .} \tag{6.84}$$

This result was obtained for $N\to\infty$ and after approximating the distributions of the Q variates by normal distributions. Please note that this static result contains r as an independent variable. Equation (6.84) stands for the progress rate for a given r.

6.3.2.3 Interpreting the result

Equation (6.84) can be used for the static evaluation of the progress rate. For the stationary evaluation, the value of $R^{(\infty)}$ should be inserted in r. For the parabolic ridge, it will be derived in Section 6.4.

An important information is obtained from (6.84) for ridge functions

$$0 \le \varphi \le c_{1,\lambda}\sigma\ . \tag{6.85}$$

The upper limit is easy to identify for $\alpha = 0$, or for $\alpha < 0$ *and* $r \gg 1$. The lower limit is obtained for $\alpha > 1$ *and* $r \gg 1$, for $\alpha > 2$ *and* $r \approx R^{(\infty)}$ *and* $\sigma\to\infty$. The stationary value $R^{(\infty)}$ for $\sigma\to\infty$ can be found by considering the discussion in Point 6.3.1.1. These cases will be investigated in depth in the chapter for simulations, Section 7.2.

It is important to note that the progress rate does never become negative for ridge functions. However, which value does it get for $\alpha < 2$ as $\sigma\to\infty$? An estimate for the stationary case can be obtained using the argumentation in Point 6.3.1.1. Since $R^{(\infty)}$ is

$\mathcal{O}(\sigma)$, the progress rate goes to infinity in this case, instead of becoming zero. This result is in contrast to the *evolution window* hypothesis (see Subsection 5.2.1). The ridge function is convex for $\alpha > 0$; remarkably, the stationary progress rate also goes to infinity for some convex ridge functions (with $0 < \alpha < 2$) as the mutation strength goes to infinity.

If one considers the general result in (6.84), it becomes clear that ridge function do not obey the *universal progress law* (see Subsection 5.2.4): This formula does not contain a term with negative sign which is expected to be of $\mathcal{O}(\sigma^2)$. Additionally, the hyperplane test function ($\alpha = 0$ case) seems to give the maximum φ among ridge functions (see (6.84) and (5.22)). The ridge functions with $\alpha < 0$ cannot attain a larger progress rate since the denominator cannot be smaller than one. Therefore, the hyperplane function should be considered as the *limit* fitness function for the case when the population is far from the optimum. This result contradicts to the assertion on the corridor model in Subsection 5.2.5.

Equation (6.84) represents an interesting example for the evolutionary progress principle (EPP, see Subsection 5.2.6). The two components for the progress toward optimum and perpendicular to the progress direction can be identified in the denominator of this equation. The denominator S_Q is composed of $\|\mathbf{a}\|^2$ and $\mathrm{Tr}[\mathbf{Q}^2]$ as shown in (6.18). The magnitude of vector $\mathbf{a}$ is given in (6.10). If one traces S_Q back, the gain part of the EPP emerges as the component of $\mathbf{a}$ in x_0 direction, that is, as the constant "1" in the denominator. The remaining components in the denominator act as the loss part. They vanish for $\alpha = 0$, and principally for $\alpha < 0$ (concave fitness functions). Therefore, a further example for the realization of gain and loss parts of the EPP is observed for ridge functions.

The results in (6.45) and in (6.84) can simply be compared. For the stationary case, both formulae are expected to give comparable results. Equation (6.45) is obtained for the stationary case using a simple local model in Subsection 6.3.1. It does not contain the $2\sigma^2\mathrm{Tr}[\mathbf{Q}^2]$ term emerging in (6.84); the $\mathbf{Q}$ matrix in this term is obtained using second order derivatives of the local quality function approximation in Subsection 5.3.2. In other words, one can say that (6.45) is obtained using first order derivatives, whereas (6.84) considers additionally the second order derivatives.

6.3.2.4 Linear transformation of the fitness function

Since the progress rate is measured in the search space, the linear transformations of the fitness function (i.e. $k_1 \cdot F(\mathbf{x}) + k_2$; $k_1, k_2 \in \mathrm{I\!R}$) should not affect the progress rate of ES algorithms. In other words, the progress rate observed by using any of such transformations as the fitness function should be the same. Particularly, this statement should also be valid for ridge functions, and the result should always be (6.84). Such a transformation for the parabolic ridge can be found in Equation (5.34).

For the derivation of φ, the linear transformation affects the probability distributions in the fitness space; in other words, the distribution $P_1(Q)$ and the density $p(Q|z)$ (see Equation (6.66)). Since these functions are approximated normally, we are interested in the effect on the mean value and the standard deviation. These further are determined by the approximation of the local quality function using the vector $\mathbf{a}$ and the matrix $\mathbf{Q}$ (see (5.15), (5.16), (5.40), and Subsection 5.3.3 for explanations). Therefore, one has to investigate the eventual change in $\mathbf{a}$ and $\mathbf{Q}$ caused by a linear transformation of the fitness function, and later prove why these alterations do not affect φ in (6.84) at all.

It is important to notice that the addition of a scalar value to the fitness function does not change $\mathbf{a}$ and $\mathbf{Q}$ at all. Such additive values vanish in the local quality function (see the definitions in (5.15) and (5.16)). Therefore, only the case of multiplication with a constant k_1 remains to be investigated. In this case, $\|\mathbf{a}\|^2$ and $\mathrm{Tr}[\mathbf{Q}^2]$ are multiplied with k_1^2, and $\mathrm{Tr}[\mathbf{Q}]$ with k_1; therefore, M_Q and S_Q with k_1. Since Q scales with k_1, $P_1(Q)$ is not affected at all (see Equation (6.77)).

One would say that the final result in (6.84) should be k_1 times less since it contains only S_Q; however, this is not the case. A factor k comes to the numerator, caused by the additional substitution $Q' := \frac{Q}{k_1}$, $\mathrm{d}Q' := \frac{\mathrm{d}Q}{k_1}$ in (6.78). After this substitution, s can be introduced in (6.79) next without any problem, and the result obtained will be comparable with the ones obtained for $F(\mathbf{x}) = F_R(\mathbf{x})$. As a result, the progress rate values obtained for linear transformations of the general ridge function are also described by (6.84).

Linear transformation of the fitness function *affects* the quality gain $\overline{Q}$. According to the definition of $\overline{Q}$ in (4.1), the additional constant k_2 of such a transformation does not influence $\overline{Q}$. However, k_1 can be observed as a factor in the final formula (see e.g. (6.19) for the hyperplane, or consider the changes in (6.37) for the sphere model $F_2(\mathbf{x})$ and their effects in $\overline{Q}$).

6.3.2.5 The parabolic ridge

For $\alpha = 2$, the progress rate is obtained from (6.84) as

$$\boxed{\varphi = \frac{c_{1,\lambda}\,\sigma}{\sqrt{1 + (2dr)^2 + 2d^2(N-1)\sigma^2}}} \quad . \tag{6.86}$$

This equation gives the (static) progress rate on the parabolic ridge. It can be compared to the formula in (5.35) proposed by Rechenberg. He derived the progress rate formula for $r = 0$. Therefore, Equation (6.86) should be used in this comparison only after the substitution $r = 0$. Even after this substitution, it does not have a negative term of $\mathcal{O}(\sigma^2)$. Both formulae will be compared to simulation results in Section 7.2.9.

For a given σ, the progress rate φ in (6.86) is maximized at $r = 0$, i.e. on the ridge axis. This result is contradictory to the assertion of Rechenberg that the progress rate should be *small* on the ridge axis (cf. Point 5.3.5.3). One can investigate (6.86) further for the *maximal static performance* of the $(1,\lambda)$-ES. After inserting $r = 0$ in (6.86) and reordering terms, it becomes immediately clear that the maximum value is obtained for $\sigma \to \infty$

$$\hat{\varphi}_{st} = \hat{\varphi}|_{r=0} = \lim_{\sigma\to\infty} \varphi|_{r=0} = \lim_{\sigma\to\infty} \frac{c_{1,\lambda}}{\sqrt{\frac{1}{\sigma^2} + 2d^2(N-1)}} = \frac{c_{1,\lambda}}{d\sqrt{2N-2}} \quad . \tag{6.87}$$

The value of the optimal mutation strength $\hat{\sigma}$ is therefore infinity, which differs considerably form the value $\hat{\sigma} = c_{1,\lambda}c/2d\sqrt{N}$ in (5.38) proposed by Rechenberg (see Point 5.3.5.3).

In the next step, the stationary value of φ will be investigated by using the $R^{(\infty)^2}$ value to be derived in Subsection 6.4.1. The stationary φ value is the expected value of the progress rate for all possible r values. It can be obtained theoretically by considering the probability density of r in the stationary case. Strictly speaking, φ (6.86) depends on the random variable r, and we are interested in $\mathrm{E}\{\varphi\}$ in the stationary case over all possible values of r. The value $\mathrm{E}\{\varphi\}$ can be approximated by the static φ value at $R^{(\infty)}$, since $R^{(\infty)}$ is the expected value of r. The validity of this approximation will be shown by stationary experiments in Section 7.2. Therefore, the random variable r^2 in (6.86) is substituted by its expected value $R^{(\infty)^2}$. After inserting (6.165) in (6.86), one obtains

$$\varphi = \frac{c_{1,\lambda}^2}{\sqrt{\frac{c_{1,\lambda}^2}{\sigma^2} + \frac{[d(N-1)]^2}{2}\left(1 + \sqrt{1 + \left(\frac{2c_{1,\lambda}}{d(N-1)\sigma}\right)^2}\right)}} \ . \tag{6.88}$$

This result can be simplified by applying the normalizations for σ^* and φ^* in (6.48)

$$\varphi^* = \frac{c_{1,\lambda}\,\sigma^*}{\sqrt{1 + \frac{\sigma^{*2}}{2c_{1,\lambda}^2}\left(1 + \sqrt{1 + \left(\frac{2c_{1,\lambda}}{\sigma^*}\right)^2}\right)}} = \frac{c_{1,\lambda}^2}{\sqrt{\frac{c_{1,\lambda}^2}{\sigma^{*2}} + \frac{1}{2} + \frac{1}{2}\sqrt{1 + \left(\frac{2c_{1,\lambda}}{\sigma^*}\right)^2}}} \ . \tag{6.89}$$

The progress rate formulae in (6.88) and (6.89) attain their maximum value for $\sigma \to \infty$ and $\sigma^* \to \infty$, respectively. For this limit, the respective denominator is minimized. The resulting maximum values for φ and φ^* are (cf. (6.47) and (6.49))

$$\hat{\varphi} = \lim_{\sigma \to \infty} \varphi = \frac{c_{1,\lambda}^2}{d(N-1)}, \qquad \hat{\varphi}^* = \lim_{\sigma^* \to \infty} \varphi^* = c_{1,\lambda}^2 \ . \tag{6.90}$$

Similarly, one obtains

$$\lim_{\sigma \to 0} \frac{\varphi}{\sigma} = c_{1,\lambda}, \qquad \lim_{\sigma^* \to 0} \frac{\varphi^*}{\sigma^*} = c_{1,\lambda} \ . \tag{6.91}$$

These limits are to be compared with the simulation results in Section 7.2.1.

6.3.2.6 The sharp ridge

The sharp ridge is the member of the ridge function family with $\alpha = 1$ (cf. (3.15) and (3.13)). Therefore, its progress rate formula is obtained simply by substituting $\alpha = 1$ in (6.84)

$$\varphi = \frac{c_{1,\lambda}\,\sigma}{\sqrt{1 + d^2 + 2(N-2)\left(\frac{d}{2r}\right)^2 \sigma^2}} \approx \frac{c_{1,\lambda}\,\sigma}{\sqrt{1 + d^2}} \ . \tag{6.92}$$

For sufficiently large N, the third term in the denominator can be neglected if r is of $\mathcal{O}(N)$. This is for example the case for $r \approx R^{(\infty)}$ (see Equation (6.41) and Subsection 7.2.5). The resulting approximation is equal to (6.52) obtained using the local model. Moreover, this approximation is also supposed to be valid for $r \gg R^{(\infty)}$ or $r \to \infty$; therefore, the φ value does *not* become zero for this limit. For the parabolic ridge case or for $\alpha > 2$, the progress rate goes to zero for $r \to \infty$. As a result, a different static progress behavior is predicted for the sharp ridge (see Subsection 7.2.10).

Two final remarks should be added on the normalization of the progress rate of the sharp ridge and on the generalization of this result to other ES algorithms. The normalization in (6.51) cannot be applied to $\alpha = 1$. On the other hand, the formula in (6.92) can be generalized to other ES algorithms by using the appropriate progress coefficient instead of $c_{1,\lambda}$. For the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES, consider the remarks in Subsection 6.3.4, Page 110.

6.3.3 The $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES

The method of induced order statistics (Subsection 5.3.7) was successfully used in the previous subsection for the analysis of the progress rate φ of the $(1, \lambda)$-ES on ridge functions. In this subsection, it will be applied to the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES. The progress rate formula to be obtained is valid for all ridge functions. However, because of the approximations in the derivation, this formula is more accurate for the parabolic ridge case (see the related notes in Subsection 6.1.1 and Point 6.3.2.1).

The derivation of the progress rate formula is based on the same idea as on the $(1, \lambda)$-ES. However, in this case, the progress rate is defined using the centroids of consecutive generations. Therefore, the progress rate definition in (4.6) becomes

$$\varphi := \mathrm{E}\{\|\hat{\mathbf{x}} - \langle\mathbf{x}\rangle^{(g)}\| - \|\hat{\mathbf{x}} - \langle\mathbf{x}\rangle^{(g+1)}\|\} \ . \tag{6.93}$$

The performance of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES algorithm is not influenced by rotations, since the mutations are isotropic, and since they are generated at the same state. As a result, the fitness function can be rotated to simplify the analysis. If the progress direction $\mathbf{v}$ of the general ridge function $F_{RR}(\mathbf{x})$ (Equation (3.17)) is aligned with the x_0 axis, i.e. if x_0 is the progress axis, (6.93) can be rewritten for ridge functions as

$$\varphi := \mathrm{E}\{\langle x_0\rangle^{(g+1)} - \langle x_0\rangle^{(g)}\} = \mathrm{E}\{\langle \Delta x_0\rangle^{(g)}\} \ , \tag{6.94}$$

i.e. the average x_0 value is expected to increase over generations. This equation is the starting point for the progress rate analysis of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES.

It is important to note that the analysis of the $(\mu/\rho_{\mathrm{I}}, \lambda)$-ES is much more complicated if $1 < \rho < \mu$. As already mentioned in Subsection 2.4.1, the next generation is created by mutations applied to the same common centroid $\langle\mathbf{x}\rangle^{(g)}$ for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES. This simplifies the analysis of the algorithm considerably. For the general case $1 < \rho < \mu$, the intermediate recombination of ρ parents (the centroid formulation) generally does not give the same state in the search space. As a result, the mutation operator is applied on different points. Consequently, one has to estimate the distribution of these centroids of ρ parents each.

The number of such potential centroids is much larger that μ, but the distribution of them could be estimated using the parental distribution. Finally, the analysis of the $(\mu/\rho_{\mathrm{I}}, \lambda)$-ES is expected to be at least as complicated as the one of the (μ, λ)-ES, which does not use recombination. The analysis of the (μ, λ)-ES was done on the sphere model in [Bey95b], [Bey96c, Chapter 5]. It is much more complicated than the analysis of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES. Therefore, the analysis of both (μ, λ)-ES and $(\mu/\rho_{\mathrm{I}}, \lambda)$-ES will be avoided in scope of this work.

In [Bey95a], [Bey96c, pp. 220-221], it was shown using plausibility arguments on the sphere model that the performance of the $(\mu/\rho_{\mathrm{I}}, \lambda)$-ES is maximized if $\rho = \mu$. This value is found as a compromise: To minimize the loss term, ρ should be as large as possible; i.e. $\rho = \lambda$, which implies $\mu = \lambda$ and $\varphi = 0$. To maximize the gain term, one has to choose $\rho = \mu = 1$, by choosing the best individual only. However, for $\rho = 1$, one has neither recombination nor genetic repair. The value $\rho = \mu$ appears reasonably the best choice, where this result may be function dependent. It also assumes a certain selection ratio. For instance, the optimal ρ value of the $(9/\rho_{\mathrm{I}}, 10)$-ES on the parabolic ridge is *not* $\rho = 9$.

The analysis of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES starts with Equation (6.94). The x_0 components can be averaged for generation g and $g+1$, giving average values for successive generations. The description of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES can be found in Subsection 2.4.1. In this algorithm, the offspring are generated by applying isotropic mutations on the centroid $\langle \mathbf{x} \rangle^{(g)}$. The best μ of them are selected as the next generation to yield $\mathbf{P}^{(g+1)}$. The progress can simply be measured on mutation vectors, that is, as the average of the expected values of the components of the μ best individuals. In other words, one has to determine the μ best descendants, and later the components of generating mutations in the progress direction. Thereafter, these components should be averaged in the last step. If z denotes the component of a mutation in the direction toward the optimum, and $z_{m;\lambda}$ the corresponding component of the m-th best individual, (6.94) can be specified further

$$\varphi := \mathrm{E}\{\langle z \rangle\} = \mathrm{E}\left\{ \frac{1}{\mu} \sum_{m=1}^{\mu} z_{m;\lambda} \right\} = \frac{1}{\mu} \sum_{m=1}^{\mu} \mathrm{E}\{z_{m;\lambda}\} \ . \tag{6.95}$$

The probability density of $z_{m;\lambda}$ can be denoted as $p_{m;\lambda}(z)$, which will be described next. The expected value in (6.95) is calculated by integrating over all possible values of $z_{m;\lambda}$. Therefore one obtains

$$\varphi = \frac{1}{\mu} \sum_{m=1}^{\mu} \int_{-\infty}^{\infty} z \, p_{m;\lambda}(z) \mathrm{d}z \ . \tag{6.96}$$

Since the descendant having $z_{m;\lambda}$ has the m-th best fitness value among λ offspring, $m-1$ of the offspring should have better fitness values, and $\lambda - m$ of them worse. The number of possible constellations for the m-th best one should be calculated at this point. For λ offspring, $\lambda!$ different orderings are possible. Since we are interested only for the m-th one, the partial orderings of $m-1$ better individuals or $\lambda - m$ worse individuals are irrelevant. Similarly, one obtains $(m-1)!$ and $(\lambda - m)!$ different orderings for these subranks. Since these different cases are irrelevant for the m-th best, the number of different constellations for the m-th best is obtained as $\lambda!/(m-1)!(\lambda-m)!$. The $p_{m;\lambda}(z)$ density can be seen as the product of this number, of the mutation density $p_z(z)$, and of the acceptance probability $P_{a\,m;\lambda}(z)$ of the mutant generated as the m-th best one. As a result, $p_{m;\lambda}(z)$ reads

$$p_{m;\lambda}(z) = \frac{\lambda!}{(m-1)!(\lambda-m)!}\, p_z(z)\, P_{a\,m;\lambda}(z) \ . \tag{6.97}$$

This equation is analogous to (6.63) for the $(1,\lambda)$-ES. The number of constellations as well as the acceptance probability differ for these two algorithms. However, the mutation distribution $p_z(z)$ (Equation (5.4)) is the same for both cases.

In the next step, the acceptance probability distribution $P_{a\,m;\lambda}(z)$ will be derived. A descendant is accepted as the m-th best among all λ descendants if $\lambda-m$ of them are worse and $m-1$ are better. The distribution of "being worse" is described for a single descendant by the cdf $P_1(Q)$, and it was used in (6.64) for the acceptance probability distribution of the $(1,\lambda)$-ES. Conversely, the distribution of "being better" is described by $1-P_1(Q)$. There are $m-1$ better individuals with the distribution $[1-P_1(Q)]^{m-1}$, and $\lambda-m$ worse with the distribution $[P_1(Q)]^{\lambda-m}$. The parameter $Q_{|z}$ should be used for these expressions to indicate that these densities are calculated for an LQF value for a given z. The density $p(Q_{|z}|z)$ describes the distribution of $Q_{|z}$ conditional to a given z. If one integrates the product of these three quantities over all possible $Q_{|z}$ values, one obtains the acceptance probability for a single constellation

$$P_{a\,m;\lambda}(z) = \int_{-\infty}^{\infty} p(Q_{|z}|z)\, [P_1(Q_{|z})]^{\lambda-m}\, [1-P_1(Q_{|z})]^{m-1}\, \mathrm{d}Q_{|z} \ . \tag{6.98}$$

The progress rate φ can be expressed by combining the partial results in (6.96), (6.97), and (6.98)

$$\begin{aligned} \varphi &= \frac{1}{\mu}\sum_{m=1}^{\mu}\int_{-\infty}^{\infty} z\, \frac{\lambda!}{(m-1)!(\lambda-m)!}\, p_z(z)\, P_{a\,m;\lambda}(z)\, \mathrm{d}z \\ &= \frac{\lambda!}{\mu}\sum_{m=1}^{\mu}\int_{-\infty}^{\infty} z\, p_z(z) \int_{-\infty}^{\infty} p(Q_{|z}|z)\, \frac{[P_1(Q_{|z})]^{\lambda-m}\,[1-P_1(Q_{|z})]^{m-1}}{(m-1)!(\lambda-m)!}\, \mathrm{d}Q_{|z}\, \mathrm{d}z. \end{aligned} \tag{6.99}$$

In order to be able to integrate this expression, the integration order should be exchanged first. Thereafter, the components containing $P_1(Q_{|z})$ can be taken out of the z integral. The procedure is similar to the exchange from (6.65) to (6.66). After the substitution $Q := Q_{|z}$, one obtains

$$\varphi = \frac{\lambda!}{\mu}\int_{-\infty}^{\infty}\left(\int_{-\infty}^{\infty} z\, p_z(z)\, p(Q|z)\, \mathrm{d}z\right)\sum_{m=1}^{\mu}\frac{[P_1(Q)]^{\lambda-m}\,[1-P_1(Q)]^{m-1}}{(m-1)!(\lambda-m)!}\, \mathrm{d}Q \ . \tag{6.100}$$

Since all mutations are generated from the same point (the centroid $\langle\mathbf{x}\rangle^{(g)}$), some earlier results for the $(1,\lambda)$-ES in Subsection 6.3.2 can be used in the derivation. For example, the density $p(Q_{|z}|z)$ of the LQF values conditional to a given mutation in the direction toward optimum is also described by (6.73) for the $(\mu/\mu_I, \lambda)$-ES case. This is also true for the $P_1(Q)$ value calculated using the double integral in (6.67), since the same integral is also valid here. Therefore, the result for $P_1(Q)$ in (6.77) can be used immediately without any change. Furthermore, the parameters in these equations (M_Q, S_Q, etc.) are also applicable here, since the local quality function based on the centroid is identical to the one for the $(1, \lambda)$-ES. The parameter values can be obtained from (6.17), (6.18), (6.71), and (6.72).

The inner integral of (6.100) will be evaluated first. Inserting the definition (5.4) of $p_z(z)$ and (6.73) of $p(Q|z)$, one recognizes that this integral is identical to the inner integral for the $(1, \lambda)$-ES case in (6.78). After the substitutions (6.79), and considering (6.80) for $p(Q|z)$, the progress rate formula reads

$$\varphi = \frac{\lambda!}{\sqrt{2\pi}\mu} \frac{\sigma S_Q}{S_{Q|z}} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t\, \mathrm{e}^{-\frac{1}{2}t^2} \mathrm{e}^{-\frac{1}{2}(at+b(s))^2} \mathrm{d}t \sum_{m=1}^{\mu} \frac{[\Phi(s)]^{\lambda-m} [1-\Phi(s)]^{m-1}}{(m-1)!(\lambda-m)!} \,\mathrm{d}s. \quad (6.101)$$

The inner integral was already calculated in (6.82). Inserting the result into (6.101), one gets

$$\varphi = \frac{\lambda!}{\sqrt{2\pi}\mu} \frac{\sigma^2}{S_Q} \int_{-\infty}^{\infty} s\, \mathrm{e}^{-\frac{1}{2}s^2} \sum_{m=1}^{\mu} \frac{[\Phi(s)]^{\lambda-m} [1-\Phi(s)]^{m-1}}{(m-1)!(\lambda-m)!} \,\mathrm{d}s \ . \quad (6.102)$$

Using the following equality [AS84, p. 83], [Bey95b, p. 388], [Bey96c, p. 145]

$$\sum_{m=1}^{\mu} \frac{P^{m-1} [1-P]^{\lambda-m}}{(m-1)!(\lambda-m)!} = \frac{1}{(\lambda-\mu-1)!(\mu-1)!} \int_0^{1-P(s)} x^{\lambda-\mu-1}(1-x)^{\mu-1} \,\mathrm{d}x \ , \quad (6.103)$$

for $P = 1 - \Phi(s)$, and $\binom{\lambda}{\mu} = \lambda!/\mu!(\lambda-\mu)!$, (6.102) becomes

$$\varphi = \frac{\sigma^2}{S_Q} \frac{\lambda-\mu}{\sqrt{2\pi}} \binom{\lambda}{\mu} \int_{-\infty}^{\infty} s\, \mathrm{e}^{-\frac{1}{2}s^2} \int_0^{\Phi(s)} x^{\lambda-\mu-1}(1-x)^{\mu-1} \,\mathrm{d}x \,\mathrm{d}s \ . \quad (6.104)$$

The integrand of the inner integral is independent of s. For the inner integral, s occurs only in the upper integration limit. The outer integral is taken for all possible values of s. It is reasonable to exchange the integration order; and also adapt the integration limits properly. In (6.104), the inner integral is taken from zero to $\Phi(s)$ for a given value of s, where s is dictated by the outer integral from $-\infty$ to ∞. Hence, the limits for a double integral describe an *area*. For this special case, this area can be imagined on a plane spanned by the horizontal axis s and vertical axis x. The area is bounded by the inner integration limits $x = 0$ and $x = \Phi(s)$, and the outer integral over s is taken from $-\infty$ to ∞. *After* changing the integration order, the outer integral is to be taken over all possible values of x, i.e. from $\Phi(-\infty) = 0$ to $\Phi(\infty) = 1$. The inner integral is taken from $s = \Phi^{-1}(x)$ to $s = \infty$, to describe the same integration area. The result reads

$$\varphi = \frac{\sigma^2}{S_Q} \frac{\lambda-\mu}{\sqrt{2\pi}} \binom{\lambda}{\mu} \int_0^1 x^{\lambda-\mu-1}(1-x)^{\mu-1} \int_{\Phi^{-1}(x)}^{\infty} s\, \mathrm{e}^{-\frac{1}{2}s^2} \,\mathrm{d}s \,\mathrm{d}x \ . \quad (6.105)$$

The substitution $x = \Phi(t)$, $\mathrm{d}x = \frac{1}{\sqrt{2\pi}}\mathrm{e}^{-\frac{1}{2}t^2}\mathrm{d}t$ is carried out next. The lower limit of the inner integral becomes $\Phi^{-1}(x) = \Phi^{-1}(\Phi(t)) = t$. After the execution of the inner integral

$$\int_t^\infty s\,\mathrm{e}^{-\frac{1}{2}s^2}\,\mathrm{d}s = -\mathrm{e}^{-\frac{1}{2}s^2}\Big|_t^\infty = 0 - \left(-\mathrm{e}^{-\frac{1}{2}t^2}\right) = \mathrm{e}^{-\frac{1}{2}t^2} \ , \tag{6.106}$$

Equation (6.105) reads

$$\varphi = \frac{\sigma^2}{S_Q}\frac{\lambda-\mu}{2\pi}\binom{\lambda}{\mu}\int_{-\infty}^{\infty}\mathrm{e}^{-t^2}\left[\Phi(t)\right]^{\lambda-\mu-1}\left[1-\Phi(t)\right]^{\mu-1}\mathrm{d}t \ . \tag{6.107}$$

Taking the $c_{\mu/\mu,\lambda}$ definition (5.21) and the S_Q definition (6.18) into account, the static progress rate φ of the $(\mu/\mu_\mathrm{I},\lambda)$-ES on the ridge functions is obtained (conditional to r)

$$\boxed{\varphi = \frac{c_{\mu/\mu,\lambda}\sigma^2}{S_Q} = \frac{c_{\mu/\mu,\lambda}\sigma}{\sqrt{1+\left(d\alpha r^{\alpha-1}\right)^2 + 2\sigma^2\left[N-2+(\alpha-1)^2\right]\left(\frac{d\alpha}{2}r^{\alpha-2}\right)^2}}} \ . \tag{6.108}$$

This result was obtained for $N\to\infty$ and after approximating the distributions of the Q variates by normal distributions. This result is exact for $N\to\infty$ and $\alpha=2$. For $\alpha\neq 2$, some error is expected caused by the normal approximation of $P_1(Q)$ and $p(Q|z)$. This formula is still useful for finite N; however, it should show larger approximation error than the $(1,\lambda)$-ES formula for the same N.

Equation (6.108) can be compared easily with the progress rate formula in (6.84) obtained for the $(1,\lambda)$-ES, $\varphi = c_{1,\lambda}\sigma^2/S_Q$. The only difference is observed on the progress coefficient used. Since $c_{1,\lambda} > c_{\mu/\mu,\lambda}$ for $\mu > 1$, one may conclude that the progress rate of the $(\mu/\mu_\mathrm{I},\lambda)$-ES is *smaller* than the one of the $(1,\lambda)$-ES. Actually, this is true for the static case, i.e. if the r values are the same. For the $(\mu/\mu_\mathrm{I},\lambda)$-ES, the r value of the centroid should be used in the comparison, i.e. $r = \sqrt{\sum_{i=1}^{N-1}\langle x_i\rangle^2}$.

For the sphere model, the $(\mu/\mu_\mathrm{I},\lambda)$-ES gives better progress rate values than the $(1,\lambda)$-ES on the static case for certain values of μ (see Point 5.3.5.2). The result for the ridge functions obtained on the static case is interesting, since this observation for these two algorithms does not hold for the static progress rate on ridge functions. In [Bey96c, p. 218ff], it is conjectured that the $(\mu/\mu_\mathrm{I},\lambda)$-ES surpasses the $(1,\lambda)$-ES if $P_{s1} < \frac{1}{2}$. This critical P_{s1} value was stated as a prerequisite for the applicability of the genetic repair hypothesis. According to the formula (6.40) for the $(1,\lambda)$-ES on the ridge functions, the success probability values are in the interval $0 \leq P_{s1} \leq \frac{1}{2}$ for $\alpha \geq 0$ (see also Subsection 7.4.3). Since the static φ values of the $(\mu/\mu_\mathrm{I},\lambda)$-ES are less than the ones of the $(1,\lambda)$-ES on the ridge functions, the prerequisite $P_{s1} < 1/2$ does not seem to be relevant for this case. A simpler comparison can be made on the parabolic ridge using the corresponding P_{s1} formula (6.33) for the $(1,\lambda)$-ES. Considering the static case alone, one cannot verify this hypothesis on ridge functions; however, it holds for the stationary case.

The progress rate formulae of these two algorithms can also be compared for the *stationary* case ($r \approx R^{(\infty)}$). The stationary distance $R^{(\infty)}$ is analyzed only for the parabolic ridge case (see Section 6.4), and the maximum progress rate value $\hat{\varphi}$ is only known for the parabolic ridge ($\alpha = 2$). Comparing the $R^{(\infty)}$ value (6.202) with (6.166), one concludes that the $R^{(\infty)}$ value is *considerably smaller* for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES case. This is also true for $\alpha \geq 2$, as discussed in Point 6.3.1.1, Page 85 using (5.30) and (5.32); however, an analytic formula for $R^{(\infty)}$ remains to be derived for the general case. In the next chapter, in Subsection 7.1.4, one can also see that experimental $R^{(\infty)}$ values for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES are smaller than for the $(1, \lambda)$-ES for the same mutation strength, if $\mu < \lambda - 1$. The stationary progress rate value naturally depends on the $R^{(\infty)}$ value, since a lower $R^{(\infty)}$ means a smaller denominator and a higher progress rate φ.

For the parabolic ridge, the maximum stationary progress rate values of these two ES algorithms are given in (6.47) and (6.54). Comparing these, one concludes that the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES can progress faster than the $(1, \lambda)$-ES if

$$c_{1,\lambda}^2 < \mu c_{\mu/\mu,\lambda}^2 \ . \tag{6.109}$$

This condition is obtained using the approximate local model; however, the same result is obtained using asymptotically ($N \to \infty$) exact φ and $R^{(\infty)}$ formulae (cf. the limits in (6.90) and (6.114)). One obtains the $\hat{\varphi}$ value for the parabolic ridge as $\sigma \to \infty$, and for this case the difference between the exact and approximate formulae can be neglected.

In other words, to describe the use of recombination on the parabolic ridge, the inequality (6.109) can be used. For a fair comparison, the same λ value should be taken for the $(1, \lambda)$-ES and the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES. As one can see in Table 5.1 on Page 62, the $c_{\mu/\mu,\lambda}$ value decreases continuously as $\mu \to \lambda$. Therefore, the product $\mu c_{\mu/\mu,\lambda}^2$ should attain a maximum value at a certain μ for constant λ. For example, Condition (6.109) holds for $\lambda = 10$ in the interval $2 \leq \mu \leq 5$. Consequently, without a small selection ratio, the intermediate recombination yields an additional performance for the same λ over the mutation-selection scheme. This result is obtained for the parabolic ridge. Some selected results for other ridge functions ($\alpha > 2$) can be found in Subsection 7.2.6. The theoretical analysis of (6.109) is omitted in scope of this work (but, see Subsection 6.3.6). In the following, the parabolic ridge case is investigated further.

6.3.3.1 The parabolic ridge

Some analytical results for the parabolic ridge will be summarized here analogous to Point 6.3.2.5. The stationary case will be analyzed by using the analytically derived $R^{(\infty)}$ formula in (6.202). The maximum progress rate $\hat{\varphi}$ will be computed for $r = 0$ and $r \approx R^{(\infty)}$. The normalization of σ and φ is used to abstract the results where appropriate.

The progress rate formula in (6.108) becomes for the parabolic ridge case ($\alpha = 2$)

$$\boxed{\varphi = \frac{c_{\mu/\mu,\lambda}\sigma}{\sqrt{1 + (2dr)^2 + 2d^2(N-1)\sigma^2}}} \ . \tag{6.110}$$

This static result can simply be analyzed for $r = 0$ where φ gets its *maximum* value. The optimum value for this case can be obtained analogous to (6.87)

$$\hat{\varphi}_{st} = \hat{\varphi}|_{r=0} = \lim_{\sigma\to\infty} \varphi|_{r=0} = \lim_{\sigma\to\infty} \frac{c_{\mu/\mu,\lambda}}{\sqrt{\frac{1}{\sigma^2} + 2d^2(N-1)}} = \frac{c_{\mu/\mu,\lambda}}{d\sqrt{2N-2}} \ . \tag{6.111}$$

One observes that the $\hat{\varphi}_{st}$ value in (6.111) is smaller than the one in (6.87) for the $(1, \lambda)$-ES.

The maximum performance can also be computed for the stationary case. Firstly, the $R^{(\infty)}$ formula in (6.202) is inserted into (6.110), yielding

$$\varphi = \frac{\mu c^2_{\mu/\mu,\lambda}}{\sqrt{\left(\frac{\mu c_{\mu/\mu,\lambda}}{\sigma}\right)^2 + \frac{[d(N-1)]^2}{2}\left(1 + \sqrt{1 + \left(\frac{2\mu c_{\mu/\mu,\lambda}}{d(N-1)\sigma}\right)^2}\right)}} \ . \tag{6.112}$$

Using the normalization in (6.48), this expression simplifies to (cf. Equation (6.89))

$$\varphi^* = \frac{\mu c^2_{\mu/\mu,\lambda}}{\sqrt{\left(\frac{\mu c_{\mu/\mu,\lambda}}{\sigma^*}\right)^2 + \frac{1}{2} + \frac{1}{2}\sqrt{1 + \left(\frac{2\mu c_{\mu/\mu,\lambda}}{\sigma^*}\right)^2}}} \ . \tag{6.113}$$

The genetic repair hypothesis (Subsection 5.2.7) holds here since the $R^{(\infty)}$ value of the (μ, λ)-ES case (Equation (6.205)) is smaller than the $(1, \lambda)$-ES case (Equation (6.166)). As a result, the loss term in the denominator is smaller for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES case.

As in the $(1, \lambda)$-ES case, the maximal progress is obtained as the mutation strength goes to infinity. The respective limits for (6.112) and (6.113) read

$$\hat{\varphi} = \lim_{\sigma\to\infty} \varphi = \frac{\mu c^2_{\mu/\mu,\lambda}}{d(N-1)}, \qquad \hat{\varphi^*} = \lim_{\sigma^*\to\infty} \varphi^* = \mu c^2_{\mu/\mu,\lambda} \ . \tag{6.114}$$

These values were already found in (6.54). They will be compared with the simulation results in Subsection 7.2.3. They are larger than the values obtained for the $(1, \lambda)$-ES (see Equation (6.90)), although the values for the static case have shown an opposite relationship (compare (6.84) and (6.108)).

6.3.4 The $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES

The progress rate of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES is obtained for ridge functions by using the surrogate mutation model [Bey95a]. This model was already applied in Point 6.3.1.5; therefore, it will not be explained here in detail. The formula for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES is obtained by using $\sqrt{\mu}\sigma$ instead of σ as the mutation strength in the φ formula (6.108) for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES

$$\boxed{\varphi = \frac{\sqrt{\mu} c_{\mu/\mu,\lambda} \sigma}{\sqrt{1 + (d\alpha r^{\alpha-1})^2 + 2\mu\sigma^2\left[N - 2 + (\alpha-1)^2\right]\left(\frac{d\alpha}{2} r^{\alpha-2}\right)^2}} \ .} \tag{6.115}$$

This formula will be compared to simulation results in Subsection 7.2.7. It will be seen that this asymptotically exact formula is only useful as an approximation for larger values of N as compared to the formula for the $(1, \lambda)$-ES; e.g. for $N \geq 1000$ instead of $N \geq 100$.

An *interesting* property is observed for dominant recombination: The convergence behavior of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES is *influenced* by rotation. The rotation-dependence of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES was already mentioned in Point 6.3.1.5 for the parabolic ridge. For the hyperplane case, it was analyzed further in Point 6.3.1.6, with the result given in (6.59) on Page 93. As will be shown using experimental results in Subsection 7.2.4 and Subsection 7.2.7, the progress rate of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES on all ridge functions depends on how the progress vector $\mathbf{v}$ is chosen. Therefore, the progress rate formula for dominant recombination must contain the vector $\mathbf{v}$. The analytical derivation of this general formula is too complicated, and it is omitted in scope of this work. It is conjectured that the formula (6.115) is valid for the *diagonal* $\mathbf{v}$ case (cf. Equation (6.58)). The observed performance of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES is below these values for all other $\mathbf{v}$; however, a theoretical proof is pending.

6.3.4.1 The parabolic ridge

The progress rate φ of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES will be investigated here on the parabolic ridge. The stationary φ formula and the limit values for $\sigma \to \infty$ will be given. The relation to intermediate recombination will be demonstrated. It is important to note that the formulae in this paragraph are valid for the diagonal $\mathbf{v}$ case (cf. Equation (6.58)).

One obtains the desired φ formula by inserting $\alpha = 2$ in (6.115)

$$\boxed{\varphi = \frac{\sqrt{\mu} c_{\mu/\mu,\lambda} \sigma}{\sqrt{1 + (2dr)^2 + 2\mu d^2 (N-1)\sigma^2}}} \; . \tag{6.116}$$

This static φ formula gets its maximum value for $r = 0$. The asymptotic value reads

$$\hat{\varphi}_{st} = \hat{\varphi}|_{r=0} = \lim_{\sigma \to \infty} \varphi|_{r=0} = \lim_{\sigma \to \infty} \frac{c_{\mu/\mu,\lambda}}{\sqrt{\frac{1}{\mu\sigma^2} + 2d^2(N-1)}} = \frac{c_{\mu/\mu,\lambda}}{d\sqrt{2N-2}} \; . \tag{6.117}$$

For the stationary case ($r \approx R^{(\infty)}$), the φ formula is obtained by inserting the $R^{(\infty)}$ formula in (6.204) into (6.116)

$$\varphi = \frac{\mu c^2_{\mu/\mu,\lambda}}{\sqrt{\left(\frac{\sqrt{\mu} c_{\mu/\mu,\lambda}}{\sigma}\right)^2 + \frac{[d(N-1)]^2}{2}\left(1 + \sqrt{1 + \left(\frac{2\sqrt{\mu} c_{\mu/\mu,\lambda}}{d(N-1)\sigma}\right)^2}\right)}} \; . \tag{6.118}$$

The normalized formula is obtained by using (6.48) for (6.118). It can also be obtained by substituting σ^* by $\sqrt{\mu}\sigma^*$ in (6.113)

$$\varphi^* = \frac{\mu c^2_{\mu/\mu,\lambda}}{\sqrt{\left(\frac{\sqrt{\mu}c_{\mu/\mu,\lambda}}{\sigma^*}\right)^2 + \frac{1}{2} + \frac{1}{2}\sqrt{1 + \left(\frac{2\sqrt{\mu}c_{\mu/\mu,\lambda}}{\sigma^*}\right)^2}}} \,. \tag{6.119}$$

If one compares (6.119) with (6.113) for the $(\mu/\mu_I, \lambda)$-ES, one observes that they only differ in their denominators: The former one has $\sqrt{\mu}$ instead of μ in the two parentheses of the denominator. The maximum progress rate reads for (6.118) and (6.119)

$$\hat{\varphi} = \lim_{\sigma\to\infty} \varphi = \frac{\mu c^2_{\mu/\mu,\lambda}}{d(N-1)}, \qquad \hat{\varphi^*} = \lim_{\sigma^*\to\infty} \varphi^* = \mu c^2_{\mu/\mu,\lambda} \,. \tag{6.120}$$

These limits accord to the ones for the $(\mu/\mu_I, \lambda)$-ES in (6.114). In Section 7.2.7, the appropriateness of surrogate mutation model will be shown for the diagonal $\mathbf{v}$ case by using experiments.

6.3.5 The (μ, λ)-ES

The progress rate of the (μ, λ)-ES cannot be derived using the technique applied for the $(1, \lambda)$-ES and the $(\mu/\mu_I, \lambda)$-ES. Since mutations are practically generated from μ different states, the descendants are not normally distributed in the search space. Even if they are assumed to be distributed normally in the search space, the parameters of the distribution (i.e. its moments) are unknown. In [Bey95b], [Bey96c, Chapter 5], a correction term for the skew of the offspring distribution is introduced. The derivation of the progress rate for the (μ, λ)-ES after approximating the parameters of the population distribution is very lengthy for the sphere model. Such an additional approach is avoided in scope of this work. The progress rate formula for the (μ, λ)-ES is obtained by reasoning on the respective formulae for the $(1, \lambda)$-ES.

The progress rate formulae on the hyperplane (5.23) and sphere model (5.27) can be found for the (μ, λ)-ES in Subsection 5.3.5. The formulae for the (μ, λ)-ES and for the $(1, \lambda)$-ES differ just by the coefficient used. For both fitness functions, the formula for the (μ, λ)-ES can be obtained by simply substituting $c_{1,\lambda}$ by $c_{\mu,\lambda}$ in the progress rate formula for the $(1, \lambda)$-ES. Therefore, the formula for the $(1, \lambda)$-ES on ridge functions is also obtained by making the same substitution in (6.84)

$$\boxed{\varphi = \frac{c_{\mu,\lambda}\,\sigma^2}{S_Q} = \frac{c_{\mu,\lambda}\,\sigma}{\sqrt{1 + (d\alpha r^{\alpha-1})^2 + 2\sigma^2\left[N - 2 + (\alpha-1)^2\right]\left(\frac{d\alpha}{2} r^{\alpha-2}\right)^2}}} \,. \tag{6.121}$$

This result was obtained for $N \to \infty$ and after approximating the distributions of the Q variates by normal distributions. An alternative argumentation can be found in Point 6.3.1.7, where the approximate φ formula is derived using a local model.

Comparing (6.121) with (6.84), one can conclude that the (μ, λ)-ES has always a smaller progress rate than the $(1, \lambda)$-ES for the static evaluation: The denominator is identical for both cases, and in the numerator one has $c_{\mu,\lambda} < c_{1,\lambda}$ for all $\mu \geq 2$. The stationary case will be investigated for the parabolic ridge.

6.3.5.1 The parabolic ridge

The progress rate of the (μ, λ)-ES on the parabolic ridge can be obtained by using the formula for the general ridge function. By substituting $\alpha = 2$ in (6.121) one obtains

$$\boxed{\varphi = \frac{c_{\mu,\lambda}\sigma}{\sqrt{1 + (2dr)^2 + 2d^2(N-1)\sigma^2}}\ .} \tag{6.122}$$

The maximum static progress rate is obtained for $r = 0$, which is a general characteristic of ridge functions. For the (μ, λ)-ES, r is obtained for the virtual centroid of the population. The progress rate at $r = 0$ is maximized for $\sigma \to \infty$

$$\hat{\varphi}_{st} = \hat{\varphi}|_{r=0} = \lim_{\sigma\to\infty} \varphi|_{r=0} = \lim_{\sigma\to\infty} \frac{c_{\mu,\lambda}}{\sqrt{\frac{1}{\sigma^2} + 2d^2(N-1)}} = \frac{c_{\mu,\lambda}}{d\sqrt{2N-2}}\ . \tag{6.123}$$

By inserting the $R^{(\infty)}$ formula in (6.205) into (6.122), one obtains the stationary φ formula

$$\varphi = \frac{c^2_{\mu,\lambda}}{\sqrt{\frac{c^2_{\mu,\lambda}}{\sigma^2} + \frac{[d(N-1)]^2}{2}\left(1 + \sqrt{1 + \left(\frac{2c_{\mu,\lambda}}{d(N-1)\sigma}\right)^2}\right)}}\ . \tag{6.124}$$

Using the normalization in (6.48), one obtains

$$\varphi^* = \frac{c_{\mu,\lambda}\,\sigma^*}{\sqrt{1 + \frac{\sigma^{*2}}{2c^2_{\mu,\lambda}}\left(1 + \sqrt{1 + \left(\frac{2c_{\mu,\lambda}}{\sigma^*}\right)^2}\right)}} = \frac{c^2_{\mu,\lambda}}{\sqrt{\frac{c^2_{\mu,\lambda}}{\sigma^{*2}} + \frac{1}{2} + \frac{1}{2}\sqrt{1 + \left(\frac{2c_{\mu,\lambda}}{\sigma^*}\right)^2}}}\ . \tag{6.125}$$

The maximum progress rate for (6.124) and (6.125) reads

$$\hat{\varphi} = \lim_{\sigma\to\infty} \varphi = \frac{c^2_{\mu,\lambda}}{d(N-1)}, \qquad \hat{\varphi^*} = \lim_{\sigma^*\to\infty} \varphi^* = c^2_{\mu,\lambda}\ . \tag{6.126}$$

6.3.6 The progress efficiency η

The progress rate φ is a measure on the progress attained per generation. Therefore, it may not be adequate for comparisons of algorithms with different λ, since the number of function evaluations of the algorithms concerned will be different. As a result, one needs a further measure for the *efficiency* of algorithms. This measure is defined as the optimum progress rate per descendant generated [Bey96a], [Bey96c, p. 73]

$$\boxed{\eta := \frac{\max_{\sigma^*}[\varphi^*(\sigma^*)]}{\lambda} = \frac{\hat{\varphi}^*}{\lambda}\ .} \tag{6.127}$$

The original name for η given by Beyer is "fitness efficiency". However, since it is measured in the search space and defined using the progress rate, it will be called *progress efficiency* in this work.

Another necessary measure for the efficiency analysis is called *selection ratio*. It measures the ratio of the number of parents to the number of descendants as both numbers go to infinity. The selection ratio gets "smaller" if μ is decreased for a given λ. It is synonymously called truncation threshold, truncation ratio, or selection strength. Its definition reads for sufficiently large μ and λ [Bey96c, p. 27]

$$\boxed{\vartheta := \frac{\mu}{\lambda}\ ,} \qquad 0 < \vartheta < 1\ . \tag{6.128}$$

For the (μ, λ)-ES, ϑ can be investigated for a given μ. It can be investigated for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES for a given λ value. In both cases, one looks for the optimum value of the variable quantity in this ratio, with the aim of optimizing the performance of the algorithm. For the limit in (6.128), one gets two different ϑ values for these algorithms. The values obtained for the sphere model and for the parabolic ridge will be compared below.

For $\mu > 1$, the definition (6.127) must be principally extended; since the progress rate depends also on the value of μ. If one observes the progress rate formula for the (μ, λ)-ES on the sphere model (5.27) or on ridge functions (6.121), one simply notes that the algorithm with $\mu = 1$ is the most efficient one.

For the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES or for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES, however, the progress rate is *not* maximum for $\mu = 1$. If the recombination operator is applied, the optimum progress rate is obtained for a definite selection ratio, in other words for a certain value of μ –denoted by $\hat{\mu}$–, which depends on the given value of λ. Furthermore, $\hat{\mu}$ depends also on the value of N *if* it is calculated using an N-dependent progress rate formula. In summary, the η value should be calculated using $\hat{\mu}$ for a given λ. The progress rate is first maximized for σ and then for μ. The obtained μ value is optimal for a given λ; therefore, it is called $\hat{\mu}$ [Bey96a], [Bey96c, p. 224].

In [Bey96c, p. 220-224], the progress efficiency of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES is analyzed on the sphere model. Both N-dependent and asymptotic $(N \to \infty)$ formulae are used in the analysis. The results for $\hat{\mu}$, $\hat{\sigma^*}$, and $\hat{\varphi}^*$ are given using figures as well as on tables. The $(1, \lambda)$-ES and the (μ, λ)-ES are also analyzed in the same work for the N-dependent case and for $(N \to \infty)$. For the $(1, \lambda)$-ES, the algorithm with $\hat{\lambda} = 5$ is found to be the most efficient strategy. For the (μ, λ)-ES, the result $\hat{\mu} = 1$ is obtained for any λ. The asymptotic case is relevant to this work, since the analysis is carried out here for only $N \to \infty$.

For the parabolic ridge, the maximum performance is obtained for $\sigma^* \to \infty$. Using the stationary results mentioned in (6.90), (6.126), and (6.114), respectively, one obtains

$$(1,\lambda)\text{-ES} \quad : \qquad \eta = c_{1,\lambda}^2/\lambda \ , \tag{6.129}$$

$$(\mu,\lambda)\text{-ES} \quad : \qquad \eta = c_{\mu,\lambda}^2/\lambda \ , \tag{6.130}$$

$$(\mu/\mu_{\mathrm{I}},\lambda)\text{-ES} \quad : \qquad \eta = \mu c_{\mu/\mu,\lambda}^2/\lambda \ . \tag{6.131}$$

The respective progress efficiency values for the sphere model can be obtained simply by calculating the $\hat{\varphi}^*$ values for (5.26) through (5.28), and applying the definition in (6.127) thereafter. After comparing the results for the parabolic ridge and sphere model, one notes a remarkable similarity: The result for η on the sphere model becomes equal to the corresponding result in (6.129)–(6.131) after a multiplication by two. In other words, the progress efficiency formulae for the sphere model and for the parabolic ridge differ by just a scalar factor.

The consequence of this similarity in the η formulae can simply be explained. The $\hat{\varphi}^*$ values for the three above-mentioned algorithms differ by just this factor 1/2 between the sphere model and the parabolic ridge. The normalization applied is different for both cases; moreover, the optimum mutation strength σ^* is also different for both fitness functions. However, only the value of φ^* is relevant for the progress efficiency η. Consequently, the results obtained for the sphere model are *immediately applicable* to the parabolic ridge [Bey96c]:

1. The optimal $(1,\lambda)$-ES is obtained for $\hat{\lambda} = 5$. For this algorithm, the creation of more descendants per generation makes sense if and only if they can be created and evaluated in parallel.

2. The optimal (μ,λ)-ES is obtained for $\hat{\mu} = 1$. In other words, for a given λ, $\mu = \hat{\mu} = 1$ gives the most efficient (μ,λ)-ES. For a given μ, however, the most efficient λ is obtained by empirical investigations of (6.130). For $N \to \infty$, one obtains asymptotically $\hat{\vartheta} := \mu/\hat{\lambda} \approx 0.35$ even for $\mu \gtrapprox 30$.

3. If one has ϖ parallel processors, it is reasonable to choose the number of descendants as $\lambda = k\varpi$, $k \in \mathbb{N}$. For the $(\mu/\mu_{\mathrm{I}},\lambda)$-ES case, this yields the question of optimum number of parents. The $\hat{\mu}$ value depends on N. For $N \to \infty$, one obtains $\hat{\vartheta} := \hat{\mu}/\lambda \approx 0.27$ even for $\lambda \gtrapprox 30$.

4. The $(\mu/\mu_{\mathrm{I}},\lambda)$-ES with optimum selection ratio $\hat{\vartheta}$ is the most efficient one among these three ES algorithms if the same value is used for λ.

The hyperplane. A special case of ridge functions is the hyperplane function ($\alpha = 0$). For this case, $\hat{\eta}$ values can be obtained using the definition of η in (6.127) on the respective formulae in Point 5.3.5.1. In this case, the progress rate increases proportional to the mutation strength, and the proportionality constant is simply the progress coefficient of interest (cf. Point 6.3.1.6 for the exceptional case $(\mu/\mu_{\mathrm{D}},\lambda)$-ES). Since the progress rate is unbounded, the comparisons will be made using the same finite mutation strength for all algorithms. The magnitude of the σ used is unimportant in the comparison, it is arbitrarily chosen as one. The results are obtained by comparing the progress coefficients used.

The empirical studies on [Sch95, p. 127] for the $(1, \lambda)$-ES suggest $\hat{\lambda} = 2$ or $\hat{\lambda} = 3$. The $\eta = c_{1,\lambda}/\lambda$ values for these two cases are very near to each other (i.e. $\eta_{1,2} \approx 0.282$ and $\eta_{1,3} \approx 0.282$); it decreases for $\lambda \geq 4$. This result is generalizable to the (μ, λ)-ES since this algorithm gets the maximum φ value for a given λ if $\mu = \hat{\mu} = 1$ on this function. For the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES, one can obtain $\hat{\mu}$ for a given λ using simulation results [Bey95a], [Bey96c, p. 211]: The coefficient $c_{\mu/\mu,\lambda}$ is maximal if $\hat{\mu} = 1$. One observes that the $c_{\mu/\mu,\lambda}$ values monotonically decrease for $\mu \to \lambda$. This can be observed for $\lambda = 10$ in Table 5.1 (Page 62). Therefore, one has to maximize $\eta = c_{1,\lambda}/\lambda$ next, and obtains the same $\hat{\lambda}$ values attained for the $(1, \lambda)$-ES case.

The general ridge function. The $\hat{\lambda}$ value of the $(1, \lambda)$-ES depends on the ridge function of interest. For $\alpha = 0$ (hyperplane), see the previous paragraph. For $\alpha = 1$ (sharp ridge), the same result holds, since the progress rate formulae (e.g. (6.92)) of this fitness function asymptotically differ only by a scalar denominator from the progress rate formulae for the hyperplane. For $\alpha = 2$ (parabolic ridge), one observes $\hat{\lambda} = 5$ (as in the sphere model case). The values between $\hat{\lambda} = 3$ and $\hat{\lambda} = 5$ are expected in the interval $1 < \alpha < 2$ as a conjecture. Moreover, recombination is expected to have a positive effect on η for $\alpha > 0$; therefore, one may assert larger progress efficiency values for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES as compared to the $(1, \lambda)$-ES.

For $\alpha < 0$, one expects the same progress efficiency obtained for the hyperplane, since the maximal progress rates are expected to be the same (cf. Point 6.3.2.3). For $\alpha > 2$, one may speculate higher $\hat{\lambda}$ values since the search space gets more convex. For the analytical investigation, the $R^{(\infty)}$ formulae should be derived for these ridge functions. Otherwise, numerous simulations are necessary to determine the progress efficiency of different algorithms, in other words to locate the $\hat{\varphi}^*$ value for different λ and μ values.

6.3.7 Conclusions

In this section, different progress rate results have been obtained for ridge functions. For the derivation, the simple local model has been used first. The results obtained can at least be used as the first order approximation to the *stationary* progress rate of several ES algorithms on the general ridge function. Thereafter, the *static* case has been analyzed. The progress rate values of the $(1, \lambda)$-ES and the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES have been derived using induced order statistics. The result obtained for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES has been generalized to the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES using surrogate mutations. The progress rate of the (μ, λ)-ES has been attained using plausibility relations.

The results for the local model contain larger approximation errors; however, it gives sufficient results for the stationary case. The analysis continued using the stationary $R^{(\infty)}$ value for the parabolic ridge. The formulae obtained have been normalized, and asymptotic $(N \to \infty)$ values of $\hat{\varphi}$ and $\hat{\varphi}^*$ have been computed. Additionally, the case $r = 0$ has been investigated for all four ES algorithms, and the *maximum* static progress rate has been obtained for this value. The $r \to \infty$ case has also been investigated, and the *minimum* value for the static progress rate has been observed for this limit.

In the last step, the progress efficiency η –actually introduced as fitness efficiency in the literature– has been investigated, and it has been shown that the η formulae for the parabolic ridge and the sphere model differ just by a constant from each other, and that the $\hat{\lambda}$ values are the same for these if the same ES algorithm is used.

New or contradictory results. First of all, the progress rate is never negative for any member of ridge function family (cf. Equation (5.3) and Equation (5.35)). The progress rate of the hyperplane appears as a limit case (cf. Subsection 5.2.5).

The progress rate of the parabolic ridge is maximized as the mutation strength goes to infinity (cf. Subsection 5.2.1). This result for $\hat{\sigma}$ differs from the one derived by Rechenberg (cf. Point 5.3.5.3).

For the static case, the progress rate is maximized on the ridge axis, which is exactly opposite to the earlier results of Rechenberg (cf. Point 5.3.5.3). Surprisingly, the $(1,\lambda)$-ES outperforms the $(\mu/\mu_{\mathrm{I}},\lambda)$-ES in static progress rate figures. This is a new and counter-intuitive result.

For the stationary case, the genetic repair hypothesis (Subsection 5.2.7) is observed for the $(\mu/\mu_{\mathrm{I}},\lambda)$-ES on the parabolic ridge: Since this algorithm attains a smaller $R^{(\infty)}$ value than the (μ,λ)-ES and the $(1,\lambda)$-ES, its progress rate values are also larger.

A new form of the evolutionary progress principle (EPP, Subsection 5.2.6) is emerged on the progress rate formulae of ridge functions: The loss and gain terms appear together in the denominator, where again the loss part increases faster than the gain part with respect to the mutation strength σ. The loss term does not appear as a negative additive term (cf. Subsection 5.2.4). It *would* appear as a negative term if one would introduce a normalization scheme based on the logarithm of unnormalized φ. As can be seen in (6.51), the normalization for ridge functions is not of the form $\varphi^* = \ln(\varphi)$. Actually, such a negative term emerges if one expands the square root in the denominator as a Taylor series and cuts the series after the linear term. However, such an approach introduces numerical errors if x in $1/\sqrt{1+x}$ is *not* much less than one. The condition $x \ll 1$ is necessary for the series cut, and this condition is generally not fulfilled for $\alpha \geq 2$.

6.4 The distance r to the ridge axis

The quantity r appears in the formulae of success and progress measures. Therefore, the analysis of r emerges as an important prerequisite for the analysis of these measures. Additionally, its minimization was formulated as the short term goal (see Subsection 3.3.2).

This section has three parts. In the first two subsections, the state equation for r^2 is derived analytically for the $(1,\lambda)$-ES and the $(\mu/\mu_{\mathrm{I}},\lambda)$-ES, respectively. Using this equation, the quantities required for the static, stationary, and dynamic analysis can be derived. Additionally, the time constant to reach the stationary $R^{(\infty)}$ value is defined in Point 6.4.1.7. Furthermore, the applicability of the methods used is discussed. Subsection 6.4.3 contains only the $R^{(\infty)}$ results for the $(\mu/\mu_{\mathrm{D}},\lambda)$-ES and the (μ,λ)-ES obtained by plausibility arguments.

The distance r to the ridge axis can be investigated using static, dynamic, and stationary analysis (see Section 4.4 for definitions). The necessary tools will be developed here. In this section, the derivations for the $(1,\lambda)$-ES and the $(\mu/\mu_{\mathrm{I}},\lambda)$-ES start with an expected value integral of ${r^{(g+1)}}^2$ for a given ${r^{(g)}}^2$. The resulting state equation will be used for the *static* analysis. For instance, theoretical results are compared to the empirical ones for the extreme cases $r^{(0)}=0$ or $r^{(0)}\to\infty$ in Subsection 7.1.5. Furthermore, this state equation is used to calculate the *stationary* $R^{(\infty)}$ value. Additionally, it is also used to estimate the *progress measure* φ_R in the r direction (see Subsection 6.1.4). Finally, using this state equation iteratively, one obtains the *mean value dynamics* of r over the time.

The quantity r occurs in the formulae for $\overline{Q}$, P_{s1}, $P_{s\lambda}$, and φ. These formulae derived in the previous sections are sufficient for the static analysis of ES algorithms on ridge functions. For the stationary analysis, however, one requires the stationary value $R^{(\infty)}$. The quantity r is a random variable. The stationary values for P_{s1}, $P_{s\lambda}$, and φ could be obtained as expected values of the respective static formulae for the probability distribution of r. Alternatively, one obtains satisfactory results for the stationary case by substituting r with $R^{(\infty)}$ in the static formulae, since $R^{(\infty)}$ is the expected value of the random variable r for the stationary case. This approach assumes that the stationary value of these quantities ($\overline{Q}$, P_{s1}, $P_{s\lambda}$, and φ) can be approximated by their static value at $R^{(\infty)}$. The appropriateness of this assumption is verified by simulation results in Chapter 7. The applicability of $R^{(\infty)}$ approximation for ES algorithms on the parabolic ridge can be seen in Subsection 7.1.1 and Subsection 7.1.4. For stationary results on progress and success measures, this $R^{(\infty)}$ approximation is used for the $\alpha=2$ case. For $\alpha\neq 2$, the $R^{(\infty)}$ results obtained from the respective simulation will be used, yielding quite accurate results.

Naturally, some statistical approximations are made in the derivation of $R^{(\infty)}$. Unfortunately, the quality gain formula does suffer from the approximation error for the stationary $R^{(\infty)}$ *and* large σ, although it gives very accurate results for the static r. This will be justified by the strong dependence of the $\overline{Q}$ values on the given r in Section 7.3.

In summary, the $R^{(\infty)}$ value is necessary for estimating the stationary behavior of ES algorithms. In this section, the $R^{(\infty)}$ value is derived for the $(1,\lambda)$-ES and the $(\mu/\mu_{\mathrm{I}},\lambda)$-ES on the parabolic ridge. The corresponding value for the $(\mu/\mu_{\mathrm{D}},\lambda)$-ES is obtained by using the relation between dominant and intermediate recombination. For the (μ,λ)-

ES, the formula is constructed by reasoning; its accordance to simulation results will be shown in Subsection 7.1.4. All of these $R^{(\infty)}$ formulae will be validated by simulation results (Subsection 7.1.1 and Subsection 7.1.4). The theoretical results are obtained in this section for the parabolic ridge. For other ridge functions, the respective $R^{(\infty)}$ value will be obtained from simulations (see also Section 6.3.1.1 for additional explanations). Subsection 7.1.3 shows such $R^{(\infty)}$ values for the $(1, 10)$-ES on different ridge functions. The stationary value of $\overline{Q}$, P_{s1}, $P_{s\lambda}$, and φ can be estimated by inserting these empirical values in the respective formula.

6.4.1 The $(1, \lambda)$-ES

The derivation of $R^{(\infty)}$ for the $(1, \lambda)$-ES is based on concepts familiar from progress rate derivations. The aim is to determine the expected value of $r^{(g+1)^2}$ for a given $r^{(g)^2}$. Thus, one gets a state equation. If these two values are set to be equal, one can extract the $R^{(\infty)^2}$ value. The analysis is carried out for squared variables in order to be able to use a previous result from the sphere model theory. This state equation mentioned will also be used to derive several important quantities additional to $R^{(\infty)}$.

An outline of this subsection will be given here first. The first step of the derivation is to construct an expected value integral for $r^{(g+1)^2}$. The pdf in this integral for all possible r^2 values consists of the pdf at $g+1$ for a given $r^{(g)^2}$, and a cdf for its acceptance probability. That is, the best individual among λ descendants is selected for the next generation, and the $r^{(g+1)^2}$ value of this individual is concerned in the derivation. The best individual should have a better fitness value than the other $\lambda - 1$ descendants, and this fact is reflected in the cdf of the acceptance probability. The acceptance probability is defined using the pdf in the fitness space conditional to a given $r^{(g+1)^2}$ for the best individual, and the cdf for having a fitness worse (i.e. less) than a given value. After exchanging the integration order and using some integral equalities, the double integral for $r^{(g+1)^2}$ will be solved. The result is a state integral, and will be used for the static, dynamic, and stationary analysis of r. After this overview, the derivation can be carried out next.

The value of $r^{(g+1)^2}$ can be expressed as an expected value integral for a given $r^{(g)^2}$. In the derivation, $r^{(g+1)^2}$ (the r^2 value for the best descendant) will be denoted by u, and $r^{(g)^2}$ by R^2 for notational simplicity. The expected value integral is given as

$$\mathrm{E}\{r^{(g+1)^2}\} = \mathrm{E}\{u\} = \int_0^\infty u\, p_{1,\lambda}(u)\, \mathrm{d}u \ , \tag{6.132}$$

where $p_{1,\lambda}(u)$ denotes the pdf of u. The integral is taken over all possible values of u. The pdf $p_{1,\lambda}(u)$ represents the density of the best descendant. It can be expressed as the product of the pdf $p(u)$ of u for a given R^2, and the acceptance probability $P_{a\,1,\lambda}(u)$ of having the descendant with this u value as the best descendant. Since any of λ descendants can have the best (i.e. largest) fitness, this product is multiplied by λ (the number of different constellations). Hence, $p_{1,\lambda}(u)$ reads

$$p_{1,\lambda}(u) = \lambda\, p(u)\, P_{a\,1,\lambda}(u) \ . \tag{6.133}$$

The acceptance probability $P_{a\,1,\lambda}(u)$ will be determined next. It represents the cdf that all other descendants have worse fitness values than the fitness of an individual with a given u. This comparison can be done using local quality function (LQF) values, since all descendants are pro-created from the same parent. The LQF value for a given u will be denoted by $Q_{|u}$. The probability density for $Q_{|u}$ in the fitness space conditional to a given u is represented by $p(Q_{|u}|u)$. In order to be the best LQF value, $Q_{|u}$ should be better than the remaining $\lambda - 1$ LQF values. The probability distribution for having an LQF value smaller than a given $Q_{|u}$ is denoted by $P_1(Q_{|u})$. For $\lambda - 1$ individuals, the corresponding probability is $[P_1(Q_{|u})]^{\lambda-1}$. If the product of this probability and the pdf $p(Q_{|u}|u)$ is integrated over all possible values of $Q_{|u}$, one obtains the acceptance probability $P_{a\,1,\lambda}(u)$

$$P_{a\,1,\lambda}(u) = \int_{-\infty}^{\infty} p(Q_{|u}|u)\,[P_1(Q_{|u})]^{\lambda-1}\,\mathrm{d}Q_{|u} \quad . \tag{6.134}$$

After substituting (6.134) in (6.133), and (6.133) in (6.132), one obtains the detailed definition of $\mathrm{E}\{u\}$

$$\mathrm{E}\{u\} = \int_{0}^{\infty} u\,\lambda\,p(u) \int_{-\infty}^{\infty} p(Q_{|u}|u)\,[P_1(Q_{|u})]^{\lambda-1}\,\mathrm{d}Q_{|u}\,\mathrm{d}u \quad . \tag{6.135}$$

The cdf $P_1(Q_{|u})$ does not directly depend on u. Therefore, by substituting $Q := Q_{|u}$ and after changing the integration order, one gets the expression

$$\mathrm{E}\{u\} = \lambda \int_{-\infty}^{\infty} [P_1(Q)]^{\lambda-1} \underbrace{\int_{0}^{\infty} u\,p(u)\,p(Q|u)\,\mathrm{d}u}_{I(Q)}\,\mathrm{d}Q \quad . \tag{6.136}$$

The inner integral is computed first. This integral over u is denoted by $I(Q)$ since its result contains Q as variable.

6.4.1.1 The inner integral $I(Q)$

The densities $p(u)$ and $p(Q|u)$ are required for the calculation of $I(Q)$ in (6.136). The conditional pdf $p(Q|u)$ is derived first. It will be obtained using the local quality function $Q(\mathbf{z})$ and approximated using a normal distribution. The $Q(\mathbf{z})$ value is determined at the state $\mathbf{x}$ for generation g. The superscript indicating the generation counter will be omitted. The fitness value of the parent is denoted by $F(\mathbf{x})$, of its best descendant by $F(\mathbf{x}+\mathbf{z})$, respectively. The symbol $\mathbf{z}$ should denote the mutation $(z_0, z_1, \ldots, z_{N-1})^{\mathrm{T}}$ that created the best descendant. The notations R^2 and u were introduced above for ${r^{(g)}}^2$ and ${r^{(g+1)}}^2$, respectively. They will also be used here for defining $F(\mathbf{x})$ and $F(\mathbf{x}+\mathbf{z})$, respectively.

The performance of the $(1,\lambda)$-ES is independent of how the ridge axis is placed in the search space. For simplicity of derivation, the definition of the parabolic ridge in (3.16) is considered here. The result is also valid for the rotated case. The fitness values $F(\mathbf{x})$ and $F(\mathbf{x}+\mathbf{z})$ read

$$F(\mathbf{x}) = x_0 - d\sum_{i=1}^{N-1} x_i^2 = x_0 - dR^2 \ , \tag{6.137}$$

$$F(\mathbf{x}+\mathbf{z}) = x_0 + z_0 - d\sum_{i=1}^{N-1} (x_i+z_i)^2 = x_0 + z_0 - du \ . \tag{6.138}$$

The definition of the local quality function in (4.3) yields

$$Q(\mathbf{z}) := F(\mathbf{x}+\mathbf{z}) - F(\mathbf{x}) = z_0 - d(u - R^2) \ . \tag{6.139}$$

The conditional pdf $p(Q|u)$ in (6.136) is the density of $Q(\mathbf{z})$ for a given u. Therefore, u is a constant for this density. The mean value of (6.139) is obtained as $-d(u-R^2)$, since z_0 is normally distributed with mean zero and mutation strength σ, $\mathcal{N}(0,\sigma^2)$ (isotropic mutations). The standard deviation of $Q(\mathbf{z})$ is simply σ^2, caused by z_0. Therefore, $p(Q|u)$ reads

$$p(Q|u) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left[-\frac{1}{2\sigma^2}\big(Q + d(u-R^2)\big)^2\right] \ . \tag{6.140}$$

The density $p(u)$ is also estimated using normal distribution. An earlier result is used for this purpose. According to the *Central Limit Theorem* [Roh76, p. 282], the quantity $u = r^{(g+1)^2} = \sum_{i=1}^{N-1}(x_i+z_i)^2$ can be approximated accurately by a normal distribution for $N \to \infty$. Practically, this result holds for $N \gtrapprox 30$. One can use the cdf $P(u) = P(r^2)$ derived in [Bey95b, pp. 383-384], [Bey96c, pp. 104-106] for this purpose. One has to use $N-1$ instead of N, since this distribution is defined for $N-1$ dimensional subspace for the parabolic ridge. The formula reads

$$P(u) = \Phi\left(\frac{u - R^2 - (N-1)\sigma^2}{\sigma\sqrt{4R^2 + 2(N-1)\sigma^2}}\right) \ . \tag{6.141}$$

The standard deviation of this distribution is denoted by $\sigma\mathrm{S}$ to simplify the notation,

$$\mathrm{S} := \sqrt{4R^2 + 2(N-1)\sigma^2} \ . \tag{6.142}$$

After differentiating $P(u)$ with respect to u, one obtains

$$p(u) = \frac{1}{\sqrt{2\pi}\sigma\mathrm{S}}\exp\left[-\frac{1}{2\sigma^2\mathrm{S}^2}\big(u - R^2 - (N-1)\sigma^2\big)^2\right] \ . \tag{6.143}$$

The inner integral $I(Q)$ can be evaluated next. The substitution

$$t := \frac{u - R^2 - (N-1)\sigma^2}{\sigma\mathrm{S}}, \qquad \mathrm{d}u := \sigma\mathrm{S}\mathrm{d}t; \qquad \frac{u-R^2}{\sigma} = \mathrm{S}t + (N-1)\sigma \tag{6.144}$$

is applied after inserting the definitions (6.143) and (6.140) in the $I(Q)$ integral in (6.136). This substitution aims the simplification of the $p(u)$ density to $\mathcal{N}(0,1)$. One obtains

$$\begin{aligned} I(Q) &= \frac{\mathrm{S}}{2\pi}\int_{-\infty}^{\infty} t\,\mathrm{e}^{-\frac{1}{2}t^2}\exp\left[-\frac{1}{2}\left(d\mathrm{S}t+\frac{Q}{\sigma}+d(N-1)\sigma\right)^2\right]\mathrm{d}t \\ &+ \frac{R^2+(N-1)\sigma^2}{2\pi\sigma}\int_{-\infty}^{\infty}\mathrm{e}^{-\frac{1}{2}t^2}\exp\left[-\frac{1}{2}\left(d\mathrm{S}t+\frac{Q}{\sigma}+d(N-1)\sigma\right)^2\right]\mathrm{d}t. \end{aligned} \quad (6.145)$$

Please note the change in the lower integration limit $u=0$ to $t=-(R^2+(N-1)\sigma^2)/\sigma\mathrm{S}$, which becomes exactly $-\infty$ for $N\to\infty$. After introducing the quantities

$$a := d\mathrm{S}, \qquad b := \frac{Q}{\sigma}+d(N-1)\sigma\ , \quad (6.146)$$

Equation (6.145) can be written as

$$I(Q) = \frac{\mathrm{S}}{2\pi}\int_{-\infty}^{\infty} t\,\mathrm{e}^{-\frac{1}{2}t^2}\,\mathrm{e}^{-\frac{1}{2}(at+b)^2}\,\mathrm{d}t + \frac{R^2+(N-1)\sigma^2}{2\pi\sigma}\int_{-\infty}^{\infty}\mathrm{e}^{-\frac{1}{2}t^2}\,\mathrm{e}^{-\frac{1}{2}(at+b)^2}\,\mathrm{d}t\ . \quad (6.147)$$

These two integrals can be calculated using the formulae (5.10) and (5.9), respectively. One obtains

$$I(Q) = \left(\frac{\mathrm{S}}{\sqrt{2\pi}}\frac{-ab}{1+a^2}+\frac{R^2+(N-1)\sigma^2}{\sqrt{2\pi}\sigma}\right)\frac{1}{\sqrt{1+a^2}}\exp\left[-\frac{1}{2}\frac{b^2}{1+a^2}\right]\ . \quad (6.148)$$

The values of a and b can now be substituted back from (6.146). This result can be written simpler if one notices the definitions M_Q and S_Q in (6.8). For example, one obtains

$$1+a^2 = 1+d^2\mathrm{S}^2 = 1+d^2(4R^2+2(N-1)\sigma^2) = \frac{S_Q^2}{\sigma^2}\ , \quad (6.149)$$

since the symbol r in S_Q stands for $r^{(g)}$, which is denoted as R in this section. A similar simplification is obtained for

$$b = \frac{Q}{\sigma}+\frac{d(N-1)\sigma^2}{\sigma} = \frac{1}{\sigma}(Q-M_Q)\ . \quad (6.150)$$

Therefore, (6.148) becomes

$$I(Q) = \left(-\frac{d\mathrm{S}^2\sigma}{S_Q^2}(Q-M_Q)+\frac{R^2+(N-1)\sigma^2}{\sigma}\right)\frac{\sigma}{\sqrt{2\pi}S_Q}\exp\left[-\frac{1}{2}\left(\frac{Q-M_Q}{S_Q}\right)^2\right]. \quad (6.151)$$

This result can be inserted back to (6.136) for the next step.

6.4.1.2 The outer integral

The integral (6.136) can be evaluated now. The cdf $P_1(Q)$ was calculated in Point 6.3.2.1. The result in (6.77), $P_1(Q) = \Phi[(Q - M_Q)/S_Q]$, can be used here immediately. After inserting (6.151) in (6.136), and using the substitution

$$s = \frac{Q - M_Q}{S_Q}, \qquad \mathrm{d}Q = S_Q \mathrm{d}s \tag{6.152}$$

thereafter, one obtains

$$\mathrm{E}\{u\} = \frac{\lambda}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left[R^2 + (N-1)\sigma^2 - \frac{d\mathrm{S}^2\sigma^2}{S_Q}\, s \right] \mathrm{e}^{-\frac{1}{2}s^2} [\Phi(s)]^{\lambda-1} \mathrm{d}s \ . \tag{6.153}$$

Since

$$\frac{\lambda}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}s^2} [\Phi(s)]^{\lambda-1} \mathrm{d}s = \int_{-\infty}^{\infty} \frac{\mathrm{d}}{\mathrm{d}s} [\Phi(s)]^{\lambda} \mathrm{d}s = [\Phi(s)]^{\lambda} \Big|_{-\infty}^{\infty} = 1 \ , \tag{6.154}$$

the first two terms in the bracket yield $R^2 + (N-1)\sigma^2$. The third term is compared with the $c_{1,\lambda}$ definition in (5.18), and this progress coefficient is substituted appropriately. The result reads

$$\boxed{\mathrm{E}\{u\} = R^2 + (N-1)\sigma^2 - \frac{d\mathrm{S}^2\sigma}{S_Q}\, c_{1,\lambda}\sigma \ .} \tag{6.155}$$

This equation contains several substitutions. The symbols u and R^2 stand for $r^{(g+1)^2}$ and $r^{(g)^2}$, respectively. The value of S and S_Q can be found in (6.142) and (6.8). After these back-substitutions, one obtains Equation (6.168) on Page 125, which will be the starting point for the analysis of the dynamic behavior of r. The stationary value $R^{(\infty)}$ will be computed next, and the dynamics is investigated thereafter. Additionally, the expected progress in r-direction will also be estimated using this result (cf. the measure φ_R for the alternative quality gain derivation in Subsection 6.1.4). All these cases will be revisited in the next chapter for simulation results (Section 7.1).

6.4.1.3 The stationary value $R^{(\infty)}$

Equation (6.155) can be used for the determination of the stationary value $R^{(\infty)}$. For the condition

$$\mathrm{E}\{u\} \equiv \mathrm{E}\{r^{(g+1)^2}\} \stackrel{!}{=} R^{(\infty)^2} \stackrel{!}{=} r^{(g)^2} \equiv R^2 \ , \tag{6.156}$$

$\mathrm{E}\{u\}$ and R^2 cancel each other in (6.155), and the value $S_Q = \sigma\sqrt{1 + (d\mathrm{S})^2}$ (see (6.149)) can be inserted to simplify further calculations. One obtains

$$(N-1)\sigma^2 = \frac{d\mathrm{S}^2}{\sqrt{1 + (d\mathrm{S})^2}} c_{1,\lambda}\sigma \ . \tag{6.157}$$

Two further substitutions will be used in order to simplify the remaining steps. The first one is

$$X := (d\mathrm{S})^2 = d^2(4R^2 + 2(N-1)\sigma^2) \ . \tag{6.158}$$

After applying it to (6.157), the equation will be squared, and the terms are reordered. The result reads

$$[(N-1)d\sigma]^2(1+X) = c_{1,\lambda}^2 X^2 \ . \tag{6.159}$$

The second substitution

$$K := \frac{[(N-1)d\sigma]^2}{c_{1,\lambda}^2} \tag{6.160}$$

makes (6.159) a nice and simple equation

$$X^2 - K\,X - K = 0 \ . \tag{6.161}$$

This can be solved using the well-known equation for finding the roots of a second degree polynomial

$$X = \frac{K}{2} \pm \frac{1}{2}\sqrt{K^2 + 4K} \ . \tag{6.162}$$

Since X must be positive, the negative solution corresponding to the minus sign is rejected. After inserting the value X back from (6.158), one gets

$$\begin{aligned} d^2(4R^2 + 2(N-1)\sigma^2) &= \frac{K}{2}\left[1 + \sqrt{1 + \frac{4}{K}}\right] && (6.163) \\ R^2 &= \frac{K}{8d^2}\left[1 + \sqrt{1 + \frac{4}{K}}\right] - \frac{1}{2}(N-1)\sigma^2 \ . && (6.164) \end{aligned}$$

Now we can finally insert K back from (6.160), and reorder terms

$$R^2 = \left[\frac{(N-1)\sigma}{2c_{1,\lambda}}\right]^2 \left\{\frac{1}{2}\left[1 + \sqrt{1 + \left(\frac{2c_{1,\lambda}}{d(N-1)\sigma}\right)^2}\right] - \frac{2c_{1,\lambda}^2}{N-1}\right\} \ . \tag{6.165}$$

The final result is obtained after considering (6.156)

$$\boxed{R^{(\infty)} = \frac{(N-1)\sigma}{2c_{1,\lambda}} \sqrt{\frac{1}{2}\left[1 + \sqrt{1 + \left(\frac{2c_{1,\lambda}}{d(N-1)\sigma}\right)^2}\right] - \frac{2c_{1,\lambda}^2}{N-1}} \ .} \tag{6.166}$$

This result was obtained for $N \to \infty$ and after approximating the distributions of the Q variates by normal distributions. In the derivation, the lower integration limit has been extended to $-\infty$ to obtain (6.145), which is correct for $N \to \infty$. This result gives the stationary distance $R^{(\infty)}$ to the ridge axis for the parabolic ridge. For other ridge functions, the $R^{(\infty)}$ value cannot be derived using this method, since the distribution in (6.140) depends on the exponent α.

An important remark should be made here on the derivation. Actually, the derivation was made for the quantity ${r^{(g+1)}}^2$, and the result stated in (6.166) is for $R^{(\infty)}$, and not for ${R^{(\infty)}}^2$. Actually, the last step from (6.165) to (6.166) introduces an error, since the expected value was formulated for $\mathrm{E}\{{r^{(g+1)}}^2\}$ in (6.155). This error is tolerable if the variance of ${r^{(g+1)}}^2$ is small as compared to ${r^{(g+1)}}^2$ itself. A similar discussion can be found on Page 80. The accuracy of (6.166) will be shown using experimental results in Subsection 7.1.1, see also the notes in Point 6.4.1.8.

6.4.1.4 The relation to the sphere model

The relation between the quantity $R^{(\infty)}$ and the residual distance $D^{(\infty)}$ for the sphere model (cf. Point 5.3.5.2) was discussed in Point 6.3.1.1. For ridge functions other than the parabolic ridge, this relation will be shown using simulation results in Section 7.1.3. This relation between $R^{(\infty)}$ and $D^{(\infty)}$ can be formally shown using (6.166) for the parabolic ridge case. As the mutation strength σ goes to infinity, (6.166) becomes

$$\lim_{\sigma\to\infty} R^{(\infty)} = \lim_{\sigma\to\infty} \frac{(N-1)\sigma}{2c_{1,\lambda}} \sqrt{1 - \frac{2c_{1,\lambda}^2}{N-1}} \ . \tag{6.167}$$

For sufficiently large values of N, the square root factor approaches 1. After comparing the result with the $D^{(\infty)}$ value (5.30) for the $(1,\lambda)$-ES, one notes that $R^{(\infty)}$ can be approximated by $D^{(\infty)}$ and using $N-1$ instead of N. Furthermore, if the value of d is larger, this approximation is applicable even for smaller values of σ.

6.4.1.5 The mean value dynamics for r^2

Equation (6.155) can also be used for investigating the mean value dynamics of the distance $r^{(g)}$ over generations. As a special case, one can estimate how many generations it would take to attain the stationary value $R^{(\infty)}$ (or the vicinity of it), starting at a given state $r^{(0)}$. In the first step, one inserts the values of S and S_Q from (6.142) and (6.8), respectively. The result reads

$$\boxed{\mathrm{E}\left\{{r^{(g+1)}}^2\right\} = {r^{(g)}}^2 + (N-1)\sigma^2 - \frac{d(4{r^{(g)}}^2 + 2(N-1)\sigma^2)}{\sqrt{1 + d^2\left(4{r^{(g)}}^2 + 2(N-1)\sigma^2\right)}} c_{1,\lambda}\sigma \ .} \tag{6.168}$$

As a side remark, the third term contains the static progress rate formula for the parabolic ridge (see Equation (6.86)). Equation (6.168) describes a state equation for successive r

values. It can be used iteratively to estimate the expected required number of generations to attain a given $r^{(g)}$ starting from a given $r^{(0)}$. By the structure of the formula, $r^{(g)}$ must be between $r^{(0)}$ and $R^{(\infty)}$, and $r^{(0)}$ can be larger or smaller than $R^{(\infty)}$. In simulation runs, the measured $r^{(g+1)}$ value differs from the value suggested in (6.168) because of statistical fluctuations. However, the static average over many one-generation-experiments for the same $r^{(g)}$ gives results in agreement to the ones obtained using (6.168) iteratively. Such a comparison will be made in Section 7.1.6 for $r^{(0)} = 0$ over many generations.

6.4.1.6 The static analysis

The state equation (6.168) can be used to estimate the value of $r^{(g+1)}$ for a given $r^{(g)}$ value statically. For any $r^{(g)}$ value, this formula can be used for investigating the local behavior of the ES algorithm. If it is used in combination with the progress rate φ, the state in the search space at generation $g+1$ is defined for a given $(x_0^{(g)}, r^{(g)})$-tuple. More information on measuring the convergence behavior in r-direction can be found in Point 6.4.1.8.

A further important result is obtained from (6.168) for $r^{(0)} = 0$. According to the hypothesis of gradient diffusion stated in Subsection 5.2.2, the $(1,\lambda)$-ES should stay on the ridge axis and follow the local gradient (cf. the gradient vector in (6.1)). According to this hypothesis, ${r^{(1)}}^2$ should also be *zero* or at least small. By inserting $g = 0$ and $r^{(0)} = 0$ in (6.168), one obtains

$$\mathrm{E}\{{r^{(1)}}^2\} = (N-1)\sigma^2 - \frac{2d(N-1)\sigma^2}{\sqrt{1+2d^2(N-1)\sigma^2}}\, c_{1,\lambda}\sigma \ , \tag{6.169}$$

$$= (N-1)\sigma^2 \left[1 - 2c_{1,\lambda}\left(\frac{1}{d^2\sigma^2} + 2N - 2\right)^{-\frac{1}{2}}\right] \ . \tag{6.170}$$

For $\sigma d \ll 1$ or $N \gg 1$, the second term in the bracket can be neglected. For these cases, $\mathrm{E}\{{r^{(1)}}^2\}$ becomes $(N-1)\sigma^2$. Since the state equation for r was derived under the condition $(N \to \infty)$, one can notice that the value for $\mathrm{E}\{{r^{(1)}}^2\}$ is considerably different than zero.

Another method to justify the gradient diffusion hypothesis is to compare the asymptotic limits for the progress rate φ and the orthogonal progress measure φ_R (defined in Equation (6.20)) on the ridge axis ($r = 0$). According to this hypothesis, the ES algorithm should asymptotically ($\sigma \to 0$) follow the gradient. If one can estimate $\mathrm{E}\{{r^{(1)}}^2\} \approx \mathrm{E}\{r^{(1)}\}^2$, the progress measure φ_R yields

$$\varphi_R := \mathrm{E}\{r^{(g)} - r^{(g+1)}\} = -\sqrt{N-1}\sigma \ . \tag{6.171}$$

Please note that the magnitude of φ_R in (6.171) does not express a given direction; it is negative and reflects the total magnitude of $N-1$ mutation components. In other words, it reflects the divergence from the ridge axis as an increase in the distance to the ridge axis. Therefore, the magnitude of φ_R can become much larger than the value $c_{1,\lambda}\sigma$.

For sufficiently small values of σ, the progress rate φ in (6.91) can be estimated by $\lim_{\sigma\to 0}\varphi = c_{1,\lambda}\sigma$. The order of this result in N can be compared with (6.171). One obtains

$$\lim_{\sigma \to 0} \frac{\varphi_R}{\varphi} = \lim_{\sigma \to 0} -\frac{\sqrt{N-1}\sigma}{c_{1,\lambda}\sigma} = -\frac{\sqrt{N-1}}{c_{1,\lambda}} \quad . \tag{6.172}$$

Therefore, the $(1, \lambda)$-ES is not expected to follow the gradient for $r = 0$ on the parabolic ridge. One can investigate another limit condition. For the case $\sigma \to \infty$, one obtains using the φ limit in (6.87)

$$\lim_{\sigma \to \infty} \frac{\varphi_R}{\varphi} = \lim_{\sigma \to \infty} -\frac{\sqrt{N-1}\sigma}{\frac{c_{1,\lambda}}{d\sqrt{2N-2}}} = \lim_{\sigma \to \infty} -\frac{\sqrt{2}(N-1)d\sigma}{c_{1,\lambda}} = -\infty \quad . \tag{6.173}$$

According to the gradient diffusion hypothesis, at least the limit in (6.172) should be zero. However, this theoretical estimate conflicts with the hypothesis, since the orthogonal components of mutations are not negligible.

In the general case, the length of a mutation $\mathbf{z}$ can be compared with the progress rate φ. For instance, this can be done for the stationary case, where $\varphi_R = 0$ holds. According to the gradient diffusion hypothesis, φ should be comparable with the length of $\mathbf{z}$. One obtains using (5.8)

$$\mathrm{E}\{\|\mathbf{z}\|^2\} = \mathrm{E}\left\{\sum_{i=0}^{N-1} z_i^2\right\} = \sum_{i=0}^{N-1} \mathrm{E}\{z_i^2\} = N\sigma^2 \quad . \tag{6.174}$$

Therefore, one has "$\sqrt{N}\sigma$" as the expected length of a mutation for sufficiently large N. The progress rate limit for the $(1, \lambda)$-ES can be read in the definition of $c_{1,\lambda}$ in (5.18) for unit mutation strength. This algorithm cannot proceed faster than this limit in a specified direction in the search space. The progress rate on the hyperplane can be found in (5.22). This value also emerges as the maximum progress rate for ridge functions in (6.84). Using (6.174) and an arbitrary progress rate φ, one obtains

$$\frac{\varphi^2}{\mathrm{E}\{\|\mathbf{z}\|^2\}} \le \frac{c_{1,\lambda}^2\sigma^2}{N\sigma^2} = \frac{c_{1,\lambda}^2}{N} \quad . \tag{6.175}$$

Please note that this result is independent of the mutation strength σ. For $N = 1$, this result can be larger than one (see Table 5.1 on Page 62 for some selected $c_{1,\lambda}$ values). However, for $N \gg 1$, it is definitely much smaller than one. In other words, for high-dimensional search spaces, only a small component of the mutation vector $\mathbf{z}$ can be directed toward the optimum. One observes a similar ratio for the component in the gradient direction. In general, the component of $\mathbf{z}$ in a given direction is of $\mathcal{O}(1)$ in N, whereas $\|\mathbf{z}\|$ is of $\mathcal{O}(\sqrt{N})$. The reader is referred to [Bey98] for similar results on the sphere model. That work has shown that the gradient diffusion hypothesis is at least questionable.

6.4.1.7 The time scale for $R^{(\infty)}$

Equation (6.168) can be used to numerically calculate the *time constant* ω (also called *system time*) required to reach the neighborhood of $R^{(\infty)}$. The time constant is generally denoted by τ; however, it is denoted by ω in this work to avoid confusions with the self-adaptation parameter τ (see Subsection 2.5.1). For practical purposes, the number of generations required to reach an r^2 value in the interval $[kR^{(\infty)^2}, R^{(\infty)^2}/k]$, $0 < k < 1$ is of interest. Since the formula (6.168) concerns the r^2-dynamics, this interval is defined for $R^{(\infty)^2}$. The number k is chosen in accordance to the usual practice in the *linear system theory* as $1 - \frac{1}{\mathrm{e}} \approx 0.632$, where $\mathrm{e} := \exp(1) \approx 2.718$. Therefore, the interval reads

$$\left[\left(1 - \frac{1}{\mathrm{e}}\right) R^{(\infty)^2},\ \left(\frac{\mathrm{e}}{\mathrm{e}-1}\right) R^{(\infty)^2}\right] \ . \tag{6.176}$$

The lower boundary is considered in the calculations if $r^{(0)} < R^{(\infty)}$, the upper boundary otherwise. For the $r^{(0)}$ in the interval (6.176), one trivially obtains $\omega = 0$. The time constant ω gives the expected number of generations necessary to reach the corresponding boundary for a given $r^{(0)}$ value. The value of ω can numerically be estimated using the state equation (6.168). It can also be obtained using simulations after averaging several runs. In Section 7.1.6, the empirical results obtained using the state equation can be found.

6.4.1.8 The progress measure φ_R

The progress measure $\varphi_R := \mathrm{E}\{r^{(g)} - r^{(g+1)}\}$ was introduced in (6.20) to obtain an alternative derivation for the quality gain $\overline{Q}$ of the $(1,\lambda)$-ES. Equation (6.168) can be used to obtain the expected value $\mathrm{E}\{r^{(g)^2} - r^{(g+1)^2}\}$; therefore, it can be used to estimate φ_R. The error made in this estimation will be discussed next. Starting from the definition of φ_R in (6.20), one obtains

$$\varphi_R := \mathrm{E}\{r^{(g)} - r^{(g+1)}\} = r^{(g)} - \mathrm{E}\{r^{(g+1)}\} \approx r^{(g)} - \sqrt{\mathrm{E}\{r^{(g+1)^2}\}} \ . \tag{6.177}$$

If the variance of $r^{(g+1)}$ is small as compared to $\mathrm{E}\{r^{(g+1)^2}\}$, the approximation made in (6.177) is applicable. The accuracy of this approximation will be shown using experiments in Subsection 7.1.1. A theoretical proof has not been done in scope of this work.

6.4.2 The $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES

The analysis of r for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES is the most challenging subsection of this chapter. For this algorithm, r stands for the distance of the centroid to the ridge axis. As the first goal, the state equation will be derived for r^2. This state equation can be used for the same purposes as in the $(1,\lambda)$-ES case, explained in Subsection 6.4.1. This subsection aims to get the state equation and the analytic formula for the stationary $R^{(\infty)}$ value for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES. Other results like time constant, estimation error, progress measure φ_R, etc. can be obtained analogous to the $(1,\lambda)$-ES case.

This subsection has four parts. First, the considerations on the derivation of $\mathrm{E}\{r^2\}$ for the $(\mu/\mu_\mathrm{I}, \lambda)$-ES are explained. Since this value *cannot* be derived analytically, another quantity is proposed. Its relation to $\mathrm{E}\{r^2\}$ is shown, the error made in this relation is discussed. The derivation of this alternative quantity is the second step. Third, $\mathrm{E}\{r^2\}$ is obtained using this result. This part additionally mentions some important quantities which can be obtained using the state equation for $\mathrm{E}\{r^2\}$. Lastly, this state equation is used to get the stationary value $R^{(\infty)}$ for the $(\mu/\mu_\mathrm{I}, \lambda)$-ES.

6.4.2.1 Preliminary considerations

The expected distance of the centroid at generation $g+1$ to the ridge axis, denoted by $r^{(g+1)}$ for simplicity, can be formulated using the respective distance $r^{(g)}$ at generation g and the mutations $\mathbf{z}$ generating the λ descendants

$$\begin{aligned}\mathrm{E}\left\{r^{(g+1)^2}\right\} &= \mathrm{E}\left\{\sum_{i=1}^{N-1} \langle x_i^{(g+1)}\rangle^2\right\} = \mathrm{E}\left\{\sum_{i=1}^{N-1}\left(\frac{1}{\mu}\sum_{m=1}^{\mu} x_{i_{m;\lambda}}^{(g+1)}\right)^2\right\} \\ &= \mathrm{E}\left\{\sum_{i=1}^{N-1}\left(\frac{1}{\mu}\sum_{m=1}^{\mu}\tilde{x}_{i_{m;\lambda}}^{(g)}\right)^2\right\} = \frac{1}{\mu^2}\mathrm{E}\left\{\sum_{i=1}^{N-1}\left(\sum_{m=1}^{\mu}\left[\langle x_i^{(g)}\rangle + z_{i_{m;\lambda}}^{(g)}\right]\right)^2\right\} . \end{aligned} \quad (6.178)$$

Naturally, the μ best descendants are concerned in this derivation, the notation "$m;\lambda$" was introduced on Page 105. As explained in Subsection 2.4.1, the mutations are applied in the $(\mu/\mu_\mathrm{I}, \lambda)$-ES to the centroid of the previous generation. The state equation must also be formulated in that way; first for the self-consistency of the equation and second for the consistency with the algorithm. However, the final expression in (6.178) is very complicated to evaluate. The order of summations should be exchanged, in order to obtain an expression for the μ best mutations. This transformation cannot be done in the method of induced order statistics. Equation (6.178) requires the calculation of $\langle x_i^{(g+1)}\rangle$ first, and not the average over the $r^{(g+1)^2}$ values of the best μ descendants.

The next approach is to start with an analytically tractable expression, and compare it with (6.178) to see the differences. To this end, the average of the distances of μ best descendants to the ridge axis is considered

$$\begin{aligned}\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \mathrm{E}\left\{\frac{1}{\mu}\sum_{m=1}^{\mu} r_m^{(g+1)^2}\right\} = \frac{1}{\mu}\mathrm{E}\left\{\sum_{m=1}^{\mu}\left(\tilde{r}_{m;\lambda}^{(g)}\right)^2\right\} \\ &= \frac{1}{\mu}\mathrm{E}\left\{\sum_{m=1}^{\mu}\sum_{i=1}^{N-1}\left(\tilde{x}_{i_{m;\lambda}}^{(g)}\right)^2\right\} = \frac{1}{\mu}\mathrm{E}\left\{\sum_{m=1}^{\mu}\sum_{i=1}^{N-1}\left[\langle x_i^{(g)}\rangle + z_{i_{m;\lambda}}^{(g)}\right]^2\right\} . \end{aligned} \quad (6.179)$$

After comparing the final expression in (6.179) to the one in (6.178), one notes that the brackets contain the same expression for both equations. Unfortunately, one observes two important differences: The order of summations is different, and the double sum is divided by μ in (6.178), and by μ^2 in (6.179). One may claim that $\mathrm{E}\{r^{(g+1)^2}\}$ and $\mathrm{E}\{\langle r^{(g+1)^2}\rangle\}$

differ very much from each other, but they do not. In the following, these two expressions will be evaluated to the end. Equation (6.179) will be evaluated first. For sake of simplicity, the superscript $^{(g)}$ will be omitted in the following

$$
\begin{aligned}
\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \frac{1}{\mu}\mathrm{E}\left\{\sum_{m=1}^{\mu}\sum_{i=1}^{N-1}\left[\langle x_i\rangle^2 + 2\langle x_i\rangle z_{i_{m;\lambda}} + z_{i_{m;\lambda}}^2\right]\right\} \\
&= \frac{1}{\mu}\mathrm{E}\left\{\sum_{m=1}^{\mu}\left[r^2 + \sum_{i=1}^{N-1}\left(2\langle x_i\rangle z_{i_{m;\lambda}} + z_{i_{m;\lambda}}^2\right)\right]\right\} \\
&= r^2 + \frac{2}{\mu}\mathrm{E}\left\{\sum_{m=1}^{\mu}\sum_{i=1}^{N-1}\langle x_i\rangle z_{i_{m;\lambda}}\right\} + \frac{1}{\mu}\mathrm{E}\left\{\sum_{i=1}^{N-1}\sum_{m=1}^{\mu} z_{i_{m;\lambda}}^2\right\} \\
&= r^2 + 2\mathrm{E}\left\{\sum_{i=1}^{N-1}\langle x_i\rangle\frac{1}{\mu}\sum_{m=1}^{\mu} z_{i_{m;\lambda}}\right\} + \mathrm{E}\left\{\sum_{i=1}^{N-1}\langle z_i^2\rangle\right\} \quad . \qquad (6.180)
\end{aligned}
$$

Please note that $r^2 = r^{(g)^2}$ stands for the r^2 value of the centroid at generation g. Therefore, it is identical for all descendants created in generation g. Similarly, for (6.178) one gradually obtains

$$
\begin{aligned}
\mathrm{E}\left\{r^{(g+1)^2}\right\} &= \frac{1}{\mu^2}\mathrm{E}\left\{\sum_{i=1}^{N-1}\left(\mu\langle x_i\rangle + z_{i_{1;\lambda}} + \cdots + z_{i_{2;\lambda}} + z_{i_{\mu;\lambda}}\right)^2\right\} \\
&= \frac{1}{\mu^2}\mathrm{E}\left\{\sum_{i=1}^{N-1}\left(\mu^2\langle x_i\rangle^2 + 2\mu\langle x_i\rangle\sum_{m=1}^{\mu} z_{i_{m;\lambda}} + \sum_{m=1}^{\mu} z_{i_{m;\lambda}}^2 + \sum_{m=1}^{\mu}\sum_{\substack{k=1\\k\neq m}}^{\mu} z_{i_{m;\lambda}} z_{i_{k;\lambda}}\right)\right\} \\
&= r^2 + 2\mathrm{E}\left\{\sum_{i=1}^{N-1}\langle x_i\rangle\frac{1}{\mu}\sum_{m=1}^{\mu} z_{i_{m;\lambda}}\right\} + \frac{1}{\mu}\mathrm{E}\left\{\sum_{i=1}^{N-1}\langle z_i^2\rangle\right\} + T_4 \quad . \qquad (6.181)
\end{aligned}
$$

In the last step of the derivation, the fourth term T_4 is assumed to be zero. It is necessary to investigate this assumption in more detail. The components z_i of mutation vectors are normally distributed ($z_i \sim \mathcal{N}(\langle x_i\rangle, \sigma^2)$). Strictly speaking, this is unfortunately not the case after selection; but the error introduced by this assumption is expected to be asymptotically ($N \to \infty$) negligible.

For the analysis of r, the $N-1$ components of the mutation vector $\mathbf{z}$ are relevant, i.e. we are interested in the subvector $(z_1, z_2, \ldots, z_{N-1})^{\mathrm{T}}$ denoted by $\mathbf{z}'$. In other words, we are interested in the component in the direction of the $(N-1)$ dimensional vector $\mathbf{e}_r$ which has been introduced in Point 6.3.1.2. For a more rigorous treatment, one has to decompose the $(N-1)$-dimensional subvector $\mathbf{z}'$ of the mutation vector $\mathbf{z}$ in two components. The transformation to a local coordinate system is necessary for that. One component will be in the $\mathbf{e}_R$ direction (the radial progress direction) and will have the magnitude y. The other component (denoted by $\mathbf{h}$) will cover the remaining $(N-2)$ directions that are selection-invariant, i.e. $\mathbf{z}' = -(\mathbf{e}_r^{\mathrm{T}}\mathbf{z})\mathbf{e}_r = y\mathbf{e}_R + \mathbf{h}$. One obtains for the fourth term of

(6.181)

$$
\begin{aligned}
T_4 &= \frac{1}{\mu^2}\mathrm{E}\left\{\sum_{i=1}^{N-1}\sum_{m=1}^{\mu}\sum_{\substack{k=1\\k\neq m}}^{\mu} z_{i_{m;\lambda}} z_{i_{k;\lambda}}\right\} = \frac{1}{\mu^2}\mathrm{E}\left\{\sum_{m=1}^{\mu}\sum_{\substack{k=1\\k\neq m}}^{\mu} \mathbf{z}'^{\mathrm{T}}_{m;\lambda}\mathbf{z}'_{k;\lambda}\right\}\\
&= \frac{1}{\mu^2}\sum_{m=1}^{\mu}\sum_{\substack{k=1\\k\neq m}}^{\mu} \mathrm{E}\left\{(y_{m;\lambda}\mathbf{e}_R + \mathbf{h}_{m;\lambda})^{\mathrm{T}}(y_{k;\lambda}\mathbf{e}_R + \mathbf{h}_{k;\lambda})\right\}\\
&= \frac{1}{\mu^2}\sum_{m=1}^{\mu}\sum_{\substack{k=1\\k\neq m}}^{\mu} \mathrm{E}\left\{y_{m;\lambda}y_{k;\lambda}\right\} + \frac{1}{\mu^2}\sum_{m=1}^{\mu}\sum_{\substack{k=1\\k\neq m}}^{\mu} \mathrm{E}\left\{\mathbf{h}^{\mathrm{T}}_{m;\lambda}\mathbf{h}_{k;\lambda}\right\} + 0 \ . \qquad (6.182)
\end{aligned}
$$

The third term (0) in (6.182) represents the other two terms obtained from the dot product. It is zero since $\mathbf{e}_R$ and the respective $\mathbf{h}$ are orthogonal to each other. The second term in (6.182) is zero since the $\mathbf{h}$ components of two mutations are uncorrelated and have the mean value zero. The first component is of $\mathcal{O}(\sigma^2)$ and of $\mathcal{O}(1)$ in N. It can asymptotically $(N \to \infty)$ be neglected as compared to the third term of (6.181) which is $\mathcal{O}(\sigma^2)$ and $\mathcal{O}(N)$. As a result, the fourth term of (6.181) is asymptotically $(N \to \infty)$ negligible. The expected value $\mathrm{E}\{\sum_{i=1}^{N-1}\langle z_i^2\rangle\}$ is calculated using induced order statistics similar to the derivation of φ or $\mathrm{E}\{\langle r^{(g+1)^2}\rangle\}$ (Appendix A). It will be approximated by $(N-1)\sigma^2$. This value is asymptotically $(N \to \infty)$ the length of the $(N-1)$-dimensional average vector of μ mutations *before* selection (see also (6.174)). Please note that this expected value also occurs in (6.180). After these considerations, one obtains by subtracting (6.180) from (6.181)

$$
\mathrm{E}\left\{r^{(g+1)^2}\right\} \simeq \mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} - \left(1 - \frac{1}{\mu}\right)(N-1)\sigma^2 \ . \qquad (6.183)
$$

Therefore, one immediately gets the desired quantity $\mathrm{E}\left\{r^{(g+1)^2}\right\}$ if (6.180) can be obtained analytically.

6.4.2.2 Derivation of $\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\}$

The descendants are generated in the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES by mutations applied at the same state, at the centroid of the population. Therefore, there are similarities to the $(1, \lambda)$-ES in the analysis. Since the $(1, \lambda)$-ES generates its descendants by mutations applied to the common parental state, some probability distributions can be immediately adopted from Subsection 6.4.1.

The expected value $\mathrm{E}\{\langle r^{(g+1)^2}\rangle\}$ can be rewritten as the sum of expected r^2 values of the best μ descendants. This can be accomplished by using an expected value integral, where the random variable u representing r^2 is integrated over all possible values for r^2. In this derivation, u denotes the the random variable for the r^2 value of the m-th best descendant. The density of u can be written as $p_{m;\lambda}(u)$, for having the m-th best fitness among all λ descendants. One obtains

$$\begin{aligned}
\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \mathrm{E}\left\{\frac{1}{\mu}\sum_{m=1}^{\mu} r_m^{(g+1)^2}\right\} = \frac{1}{\mu}\sum_{m=1}^{\mu}\mathrm{E}\left\{\left(\tilde{r}_{m;\lambda}^{(g)}\right)^2\right\} \\
&= \frac{1}{\mu}\sum_{m=1}^{\mu}\int_0^{\infty} u p_{m;\lambda}(u)\,\mathrm{d}u \ . \qquad (6.184)
\end{aligned}$$

The investigation of the expected value in (6.184) is similar to the derivation of φ for the $(\mu/\mu_I, \lambda)$-ES in Subsection 6.3.3. Therefore, most of the technical part is analogous. Naturally, the probability densities of interest will be obtained from Subsection 6.4.1, since they do not differ from the ones used for the $(1, \lambda)$-ES: The centroid plays a role similar to the single parent's in the $(1, \lambda)$-ES case.

The density $p_{m;\lambda}(u)$ can be further specified using the acceptance probability distribution $P_{a\,m;\lambda}$ and the corresponding number of constellations (cf. Equation (6.97))

$$p_{m;\lambda}(u) = \frac{\lambda!}{(m-1)!(\lambda-m)!} p(u) P_{a\,m;\lambda}(u) \ . \qquad (6.185)$$

In this equation, $p(u)$ gives the density of u for a given $r^{(g)^2}$. It has already been introduced in (6.143). The distribution $P_{a\,m;\lambda}(u)$ gives the probability for having the m-th best fitness value for a given u. It can be specified further as (cf. Equation (6.98))

$$P_{a\,m;\lambda}(u) = \int_{-\infty}^{\infty} p(Q_{|u}|u)\,[P_1(Q_{|u})]^{\lambda-m}\,[1-P_1(Q_{|u})]^{m-1}\,\mathrm{d}Q_{|u} \ . \qquad (6.186)$$

The random variable $Q_{|u}$ describes the fitness for a given u. The pdf $p(Q_{|u}|u)$ is a conditional density; it is the density for the fitness value for a given u and parental state. The distribution $P_1(Q_{|u})$ reflects the probability of having a fitness value worse than $Q_{|u}$. Since $Q_{|u}$ can attain any value in the fitness space, the integration limits are chosen accordingly. Inserting (6.186) in (6.185), and (6.185) in (6.184), one obtains

$$\begin{aligned}
\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \frac{1}{\mu}\sum_{m=1}^{\mu}\int_0^{\infty} u\frac{\lambda!}{(m-1)!(\lambda-m)!}p(u) \\
&\times \int_{-\infty}^{\infty} p(Q_{|u}|u)\,[P_1(Q_{|u})]^{\lambda-m}\,[1-P_1(Q_{|u})]^{m-1}\,\mathrm{d}Q_{|u}\,\mathrm{d}u \ . \qquad (6.187)
\end{aligned}$$

The integration order is exchanged, and the substitution $Q := Q_{|u}$ is used next (cf. the step from (6.99) to (6.100))

$$\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} = \frac{\lambda!}{\mu}\sum_{m=1}^{\mu}\int_{-\infty}^{\infty}\frac{[P_1(Q)]^{\lambda-m}\,[1-P_1(Q)]^{m-1}}{(m-1)!(\lambda-m)!}\underbrace{\int_0^{\infty} u\,p(u)\,p(Q|u)\,\mathrm{d}u}_{I(Q)}\,\mathrm{d}Q. \ (6.188)$$

The distributions from Subsection 6.4.1 will be inserted one after the other. The inner integral is named $I(Q)$. It has the same structure as the inner integral which occurred in the derivation of the state equation for the $(1, \lambda)$-ES (see Equation (6.136)). Therefore, if the distributions in the integral are also identical, the result can be adopted. Since the mutations are applied on the centroid and the mutation operator is identical to the one in the $(1, \lambda)$-ES, and since the recombination operator yields the centroid for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES case, the distributions $p(u)$ and $p(Q|u)$ are also identical for these two algorithms. The density $p(u)$ is given by (6.143) and the density $p(Q|u)$ by (6.140), respectively. The corresponding result for $I(Q)$ can be found in (6.151), Page 122.

$$\begin{aligned}
\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \frac{\lambda!}{\mu}\frac{\sigma}{\sqrt{2\pi}S_Q}\sum_{m=1}^{\mu}\int_{-\infty}^{\infty}\frac{[P_1(Q)]^{\lambda-m}\,[1-P_1(Q)]^{m-1}}{(m-1)!(\lambda-m)!} \\
&\times \left[\frac{R^2+(N-1)\sigma^2}{\sigma} - \frac{d\mathrm{S}^2\sigma}{S_Q^2}(Q-M_Q)\right]\mathrm{e}^{-\frac{(Q-M_Q)^2}{2S_Q^2}}\,\mathrm{d}Q\ . \qquad (6.189)
\end{aligned}$$

The derivation continues similar to the $(1, \lambda)$-ES case in Subsection 6.4.1. After applying the substitution for s in (6.152), and considering the $P_1(Q)$ definition in (6.77), (6.189) becomes

$$\begin{aligned}
\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \frac{\lambda!}{\sqrt{2\pi}\mu}\sum_{m=1}^{\mu}\int_{-\infty}^{\infty}\frac{[\Phi(s)]^{\lambda-m}\,[1-\Phi(s)]^{m-1}}{(m-1)!(\lambda-m)!} \\
&\times \left[R^2+(N-1)\sigma^2 - \frac{d\mathrm{S}^2\sigma^2}{S_Q}\,s\right]\mathrm{e}^{-\frac{1}{2}s^2}\,\mathrm{d}s \qquad (6.190) \\
\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \frac{\lambda!}{\sqrt{2\pi}\mu}\int_{-\infty}^{\infty}\left[R^2+(N-1)\sigma^2 - \frac{d\mathrm{S}^2\sigma^2}{S_Q}\,s\right]\mathrm{e}^{-\frac{1}{2}s^2} \\
&\times \sum_{m=1}^{\mu}\frac{[\Phi(s)]^{\lambda-m}\,[1-\Phi(s)]^{m-1}}{(m-1)!(\lambda-m)!}\,\mathrm{d}s\ . \qquad (6.191)
\end{aligned}$$

Using the identity in (6.103), the sum can be converted to an integral for $P = 1 - \Phi(s)$

$$\begin{aligned}
\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} = \frac{\lambda-\mu}{\sqrt{2\pi}}\binom{\lambda}{\mu}\int_{-\infty}^{\infty}\left[R^2+(N-1)\sigma^2 - \frac{d\mathrm{S}^2\sigma^2}{S_Q}\,s\right]\mathrm{e}^{-\frac{1}{2}s^2} \\
\times \int_0^{\Phi(s)} x^{\lambda-\mu-1}(1-x)^{\mu-1}\,\mathrm{d}x\,\mathrm{d}s\ . \qquad (6.192)
\end{aligned}$$

The integration order is exchanged next with the same considerations from (6.104) to (6.105) for integrating over the same area

$$\begin{aligned}
\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \frac{\lambda-\mu}{\sqrt{2\pi}}\binom{\lambda}{\mu}\int_0^1 x^{\lambda-\mu-1}(1-x)^{\mu-1} \\
&\times \int_{\Phi^{-1}(x)}^{\infty}\left[R^2+(N-1)\sigma^2 - \frac{d\mathrm{S}^2\sigma^2}{S_Q}\,s\right]\mathrm{e}^{-\frac{1}{2}s^2}\,\mathrm{d}s\,\mathrm{d}x\ . \qquad (6.193)
\end{aligned}$$

The substitution $x = \Phi(y)$, $\mathrm{d}x = \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\frac{1}{2}y^2} \mathrm{d}y$ yields

$$\begin{aligned} \mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} &= \frac{\lambda-\mu}{2\pi}\binom{\lambda}{\mu}\int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}y^2}[\Phi(y)]^{\lambda-\mu-1}[1-\Phi(y)]^{\mu-1} \\ &\times \int_{y}^{\infty}\left[R^2+(N-1)\sigma^2-\frac{d\mathrm{S}^2\sigma^2}{S_Q}\,s\right]\mathrm{e}^{-\frac{1}{2}s^2}\,\mathrm{d}s\,\mathrm{d}y\ . \end{aligned} \tag{6.194}$$

The result of the inner integral reads

$$\int_{y}^{\infty}[\ldots]\,\mathrm{e}^{-\frac{1}{2}s^2}\,\mathrm{d}s = \sqrt{2\pi}\left[R^2+(N-1)\sigma^2\right](1-\Phi(y)) - \frac{d\mathrm{S}^2\sigma^2}{S_Q}\left(0+\mathrm{e}^{-\frac{1}{2}y^2}\right)\ . \tag{6.195}$$

The outer integral in (6.194) can be taken separately for these two terms. One obtains using the definition of $e^{\alpha,\beta}_{\mu,\lambda}$ in (5.20)

$$\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} = \left[R^2+(N-1)\sigma^2\right]e^{0,0}_{\mu,\lambda} - \frac{d\mathrm{S}^2\sigma^2}{S_Q}e^{1,0}_{\mu,\lambda}\ . \tag{6.196}$$

The coefficient $e^{0,0}_{\mu,\lambda}$ gives the value 1. This can be proven by considering $\int_{-\infty}^{\infty} p_{m;\lambda}(u)\mathrm{d}u$ for $p_{m;\lambda}(u)$ in (6.185) and the conversion of the sum to the corresponding integral using the equality given in (6.103). After these steps and the evaluation of the inner integral, one gets $e^{0,0}_{\mu,\lambda}$. By definition, the integral $\int_{-\infty}^{\infty} p_{m;\lambda}(u)\mathrm{d}u$ must give the value 1.

The coefficient $e^{1,0}_{\mu,\lambda}$ is defined as $c_{\mu/\mu,\lambda}$ (cf. the definitions in (5.20) and (5.21)). The value of S can be substituted from (6.142), and the value of S_Q from (6.8). After all these steps and taking $R^2 = r^{(g)^2}$ into account, one obtains

$$\mathrm{E}\left\{\langle r^{(g+1)^2}\rangle\right\} = r^{(g)^2} - \frac{d\left(4r^{(g)^2}+2(N-1)\sigma^2\right)}{\sqrt{1+d^2\left(4r^{(g)^2}+2(N-1)\sigma^2\right)}}c_{\mu/\mu,\lambda}\sigma + (N-1)\sigma^2\ . \tag{6.197}$$

6.4.2.3 The derivation of $\mathrm{E}\left\{r^{(g+1)^2}\right\}$

The notation $r^{(g+1)^2}$ stands for the squared distance of the centroid to the ridge axis at generation $g+1$. Its expected value cannot be computed in a direct analytical way. However, as it was explained in Point 6.4.2.1, it can be obtained indirectly from the expected value $\mathrm{E}\{\langle r^{(g+1)^2}\rangle\}$ given by (6.197). Therefore, one can obtain an asymptotically ($N \to \infty$) exact result for $\mathrm{E}\{r^{(g+1)^2}\}$ by substituting (6.197) in (6.183). The result reads

$$\boxed{\mathrm{E}\left\{r^{(g+1)^2}\right\} = r^{(g)^2} - \frac{d\left(4r^{(g)^2}+2(N-1)\sigma^2\right)}{\sqrt{1+d^2\left(4r^{(g)^2}+2(N-1)\sigma^2\right)}}c_{\mu/\mu,\lambda}\sigma + \frac{1}{\mu}(N-1)\sigma^2.} \tag{6.198}$$

As a side remark, please note that the progress rate formula for the $(\mu/\mu_\mathrm{I}, \lambda)$-ES in (6.110) occurs as part of the second term. This state equation has a close relationship to the one of the $(1, \lambda)$-ES given in (6.168). The progress coefficients used are different, and the r-independent term is divided by the number of parents for the $(\mu/\mu_\mathrm{I}, \lambda)$-ES. Therefore, most of the derivations for the $(1, \lambda)$-ES in Subsection 6.4.1 need not to be repeated here. As for the $(1, \lambda)$-ES, the state equation (6.198) can be used to compute several quantities:

1. The derivation of the stationary value $R^{(\infty)}$ (see Point 6.4.1.3).
2. The relation of $R^{(\infty)}$ to $D^{(\infty)}$ on the sphere model (see Point 6.4.1.4).
3. The mean value dynamics of $r^{(g)}$ over generations (see Point 6.4.1.5).
4. The static evaluation of $r^{(g+1)}$ (see Point 6.4.1.6).
5. The time constant ω for a given $r^{(0)}$ (numerically, see Point 6.4.1.7).
6. The progress measure φ_R in r direction (see Point 6.4.1.8).
7. The estimation of the error made for $\mathrm{E}\left\{r^{(g+1)}\right\} \approx \sqrt{\mathrm{E}\left\{r^{(g+1)^2}\right\}}$ (see Point 6.4.1.8).

These items consider different aspects of analysis methods mentioned in Section 4.4. The value $R^{(\infty)}$ is obtained by stationary analysis. The time constant reflects an aspect of the dynamic analysis. The progress measure φ_R and the static evaluation of $r^{(g+1)}$ reflect different views of the static analysis. The stationary value $R^{(\infty)}$ will be derived next; all other items listed can be obtained using the methods described in the previous subsection (Subsection 6.4.1).

6.4.2.4 The stationary value $R^{(\infty)}$

The symbol $R^{(\infty)}$ represents the stationary distance of the centroid to the ridge axis. The derivation of $R^{(\infty)}$ is analogous to the one for the $(1, \lambda)$-ES in Point 6.4.1.3. The stationary case is obtained for Condition (6.156) on Equation (6.198). After applying the substitution X given in (6.158), the remaining terms can be written simpler. One obtains

$$\frac{1}{\mu}(N-1)\sigma^2 = \frac{X}{d\sqrt{1+X}}\, c_{\mu/\mu,\lambda}\sigma \ . \tag{6.199}$$

The aim here is to get an equation on X. Both sides of the equation can be squared, and the terms can be reordered. The substitution for L will make the following steps simpler

$$\frac{X^2}{1+X} = \left[\frac{d(N-1)\sigma}{\mu c_{\mu/\mu,\lambda}}\right]^2 =: L \ . \tag{6.200}$$

The symbol L is analogous to K introduced in (6.160) for the $R^{(\infty)}$ derivation on the $(1, \lambda)$-ES. Since the value of X is identical for both cases, the following steps after (6.161) can be adopted here. The resulting equation for ${R^{(\infty)}}^2$ can be found in (6.164). After inserting the value of L in (6.200) to K in (6.164), one obtains

$$R^{(\infty)^2} = \left[\frac{(N-1)\sigma}{2\sqrt{2}\mu c_{\mu/\mu,\lambda}}\right]^2 \left\{1 + \sqrt{1 + \left(\frac{2\mu c_{\mu/\mu,\lambda}}{d(N-1)\sigma}\right)^2}\right\} - \frac{1}{2}(N-1)\sigma^2 \ , \tag{6.201}$$

which can be rewritten as

$$\boxed{R^{(\infty)} = \frac{(N-1)\sigma}{2\mu c_{\mu/\mu,\lambda}} \sqrt{\frac{1}{2}\left[1 + \sqrt{1 + \left(\frac{2\mu c_{\mu/\mu,\lambda}}{d(N-1)\sigma}\right)^2}\right] - \frac{2\mu^2 c^2_{\mu/\mu,\lambda}}{N-1}} \ .} \tag{6.202}$$

This result was obtained for $N \to \infty$ and after approximating the distributions of the Q variates by normal distributions. In the derivation, the fourth term in (6.181) was assumed to be zero, and the third term was approximated by $\frac{1}{\mu}(N-1)\sigma^2$. Furthermore, the lower limit of the $I(Q)$ integral in (6.188) was extended to $-\infty$ after the transformation explained following Eq. (6.145) which is asymptotically correct for $N \to \infty$.

The result obtained can be approximated for larger σ and $N \to \infty$ by

$$\lim_{\sigma\to\infty} R^{(\infty)} = \lim_{\sigma\to\infty} \frac{(N-1)\sigma}{2\mu c_{\mu/\mu,\lambda}} \sqrt{1 - \frac{2\mu^2 c^2_{\mu/\mu,\lambda}}{N-1}} \ . \tag{6.203}$$

The square root gives asymptotically ($N \to \infty$) the value one. If one compares (6.203) with the $D^{(\infty)}$ value in (5.32) for $N-1$ variables, one observes asymptotically ($\sigma \to \infty$) the relation $R^{(\infty)} \approx D^{(\infty)}$. Therefore, $R^{(\infty)}$ can be approximated under these conditions by the corresponding $D^{(\infty)}$ value for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES on the sphere model. This approximation gives useful results for finite σ, whereas the lowest applicable value for σ depends also on d. If d is large, this approximation can be used starting from smaller σ values. For more information on the relation between $R^{(\infty)}$ and $D^{(\infty)}$, see Point 6.3.1.1.

6.4.3 The $R^{(\infty)}$ value for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES and (μ, λ)-ES

The value of $R^{(\infty)}$ has been derived for the $(1, \lambda)$-ES and $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES in the previous two subsections, respectively. The value for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES can be used to estimate the formula for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES. The method of surrogate mutations is used for this purpose, which was already used in Point 6.3.1.5 and Subsection 6.3.4 for the derivation of the progress rate formulae of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES. For this purpose, the mutation strength σ is substituted in (6.202) by $\sqrt{\mu}\sigma$. The result reads

$$\boxed{R^{(\infty)} = \frac{(N-1)\sigma}{2\sqrt{\mu} c_{\mu/\mu,\lambda}} \sqrt{\frac{1}{2}\left[1 + \sqrt{1 + \left(\frac{2\sqrt{\mu} c_{\mu/\mu,\lambda}}{d(N-1)\sigma}\right)^2}\right] - \frac{2\mu^2 c^2_{\mu/\mu,\lambda}}{N-1}} \ .} \tag{6.204}$$

Note that this formula is only valid if the progress vector $\mathbf{v}$ lies diagonally in the search space, as it was formalized in Equation (6.58). It is definitely not valid if $\mathbf{v}$ is aligned with a coordinate axis (Equation (6.57)).

For the (μ, λ)-ES case, the $R^{(\infty)}$ formula is obtained by substituting $c_{1,\lambda}$ by $c_{\mu,\lambda}$ in the $R^{(\infty)}$ formula (6.166) for the $(1, \lambda)$-ES. The similarity of the $D^{(\infty)}$ formulae (5.30) and (5.31) and the arguments in Point 6.3.1.1 are taken into account here. One obtains

$$\boxed{R^{(\infty)} = \frac{(N-1)\sigma}{2c_{\mu,\lambda}} \sqrt{\frac{1}{2}\left[1 + \sqrt{1 + \left(\frac{2c_{\mu,\lambda}}{d(N-1)\sigma}\right)^2}\right] - \frac{2c_{\mu,\lambda}^2}{N-1}}\ .} \tag{6.205}$$

These two results asymptotically ($\sigma \to \infty$) yield the $D^{(\infty)}$ value from the sphere model theory, provided that $D^{(\infty)}$ is taken for $N-1$ variables in (5.33) and (5.31), respectively. Their applicability for finite σ will be shown using experiments in Subsection 7.1.4. The state equations for these two ES algorithms can be obtained by applying the steps for the computation of $R^{(\infty)}$ backwards starting at these results, if necessary.

6.5 Summary and Conclusions

This chapter had four sections. They were dedicated to four convergence measures, namely quality gain $\overline{Q}$, success measures P_{s1} and $P_{s\lambda}$, progress rate φ, and distance r to the ridge axis. The longest section was the one devoted to the progress rate; it contains several results on different ridge functions for different ES algorithms. Other sections contained results for a subset of these cases. A summary of these sections and the important results will be given in the following.

6.5.1 Summary

The theoretical analysis is carried out for the measures defined in the search space (φ and r) and in the fitness space ($\overline{Q}$, P_{s1}, and $P_{s\lambda}$). The relation between these measures -which are defined in different spaces- has been searched. This analysis will also continue in the experimental chapter (Chapter 7). The determination of the search space measures is the ultimate aim of this work. Since these measures are defined directly on object variables, they have a distinguished value in the search for the optimum variable setting. Additionally, the applicability of the fitness space measures for predicting the values of the search space measures is also investigated. The analysis of the fitness space measures is to be interpreted according to that criterion.

In Section 6.1, the quality gain formula for the $(1, \lambda)$-ES has been derived on the parabolic ridge case. Two different methods have been shown for deriving this formula. The former one is based on the moments of the local quality function $Q(\mathbf{z})$. The first two moments can be used for the normal approximation of this density. These two parameters have been derived for the general case of ridge functions. The derivation of the two remaining parameters would give us the quality gain formula for the general ridge function; however, their computation is lengthy and therefore omitted in this work. Therefore, the $\overline{Q}$ formula is derived only for the parabolic ridge. The latter method is intended to determine the quality gain using progress measures in the search space. A second measure additional to the progress rate has been introduced for this purpose: The *progress measure* φ_R defines the expected decrease in the distance to the ridge axis. The resulting formula provides as a by-product the condition under which the quality gain $\overline{Q}$ gives results equivalent to the progress rate φ for ridge functions. The formula for φ_R has been obtained in Section 6.4 on the parabolic ridge. The latter method for determining $\overline{Q}$ can principally be applied to other ridge functions or other ES algorithms, too. However, the derivation of φ_R for the general ridge case is still pending.

The success measures P_{s1} and $P_{s\lambda}$ have been investigated in Section 6.2. The formulae have been obtained using the values of the first two moments of $Q(\mathbf{z})$ from Section 6.1. The parabolic ridge case has been considered first, and the formulae have been derived for the $(1, \lambda)$-ES. Additionally, the relation of the parabolic ridge to the sphere model has been established using the limit value for the success probability P_{s1}. Finally, the success probability formulae have been given for the $(1, \lambda)$-ES on the general ridge function. In Chapter 7, it will be investigated whether the optimum value of the mutation strength σ -which yields the maximum progress rate φ- can be predicted using P_{s1} values.

The progress rate φ of the general ridge function has been derived in Section 6.3 for several ES algorithms. A local model has been introduced first (Subsection 6.3.1). Using this model, the stationary value of φ has been derived for the $(1, \lambda)$-ES, the $(\mu/\mu_\mathrm{I}, \lambda)$-ES, the $(\mu/\mu_\mathrm{D}, \lambda)$-ES, and the (μ, λ)-ES on the general ridge function. These formulae require the stationary value $R^{(\infty)}$ of the distance r to the ridge axis. Therefore, an approximate formula for $R^{(\infty)}$ has been provided first. These φ formulae yield more accurate results if the actual value of $R^{(\infty)}$ is used. In the following subsections, the static progress rate value has analytically been derived for the $(1, \lambda)$-ES and the $(\mu/\mu_\mathrm{I}, \lambda)$-ES. Additionally, it has been shown that the value of the progress rate φ does not change for linear transformations of the fitness function. The static progress rate has been derived for the $(1, \lambda)$-ES in Subsection 6.3.2. Some probability distributions obtained in this subsection have been used in Subsection 6.3.3, Subsection 6.4.1, and Subsection 6.4.2. Subsection 6.3.3 investigated the progress rate φ for the $(\mu/\mu_\mathrm{I}, \lambda)$-ES. The static φ formula for the $(\mu/\mu_\mathrm{D}, \lambda)$-ES has been obtained in Subsection 6.3.4 using surrogate mutations. In Subsection 6.3.5, the (μ, λ)-ES has been considered using plausibility arguments. Additionally, the parabolic ridge case has been considered separately in Subsection 6.3.2 through Subsection 6.3.5: The respective formula has explicitly been given for static and stationary cases. The asymptotic $(\sigma \to \infty)$ values of the stationary progress rate formula have been given as well.

The *progress efficiency* η has been used in Subsection 6.3.6 to compare the performance of different ES algorithms on the parabolic ridge. The $(\mu/\mu_\mathrm{I}, \lambda)$-ES appeared to be the most efficient one of them for a large interval of the *selection ratio* ϑ. The conclusions related to the progress rate section have been summarized in Subsection 6.3.7.

Section 6.4 provided the analysis of the distance r to the ridge axis. The $(1, \lambda)$-ES has been analyzed in Subsection 6.4.1. The state equation obtained has mainly been used for four different purposes: To compute the *stationary* $R^{(\infty)}$ value, to obtain the time constant ω for $R^{(\infty)}$ numerically, for the *dynamics*, and to estimate the *static* progress measure φ_R for the alternative quality gain formula in Subsection 6.1.4. The analysis of the $(\mu/\mu_\mathrm{I}, \lambda)$-ES in Subsection 6.4.2 used some probability distributions introduced in Subsection 6.4.1. The derivation itself, however, poses another additional difficulty. The value $r^{(g+1)^2}$ of the centroid cannot directly be derived by analytical methods. Therefore, the relation between the expected value of this desired quantity and of the average of squared distances $\langle r^{(g+1)^2} \rangle$ has been shown. The analytical result for the latter has been used to approximate the former one. The state equation for the $(\mu/\mu_\mathrm{I}, \lambda)$-ES can be used for the same purposes shown in Subsection 6.4.1, and the corresponding $R^{(\infty)}$ value has been determined as an example. The $R^{(\infty)}$ value for the $(\mu/\mu_\mathrm{D}, \lambda)$-ES has been derived in Subsection 6.4.3, based on the surrogate mutation model. In the same subsection, the formula for the (μ, λ)-ES has been obtained after some reasoning.

6.5.2 Conclusions

The most important part of the analysis was devoted to the progress rate. Therefore, a more detailed overview is provided for this measure of progress in Point 6.3.1.8 (stationary results obtained using the simple local model) and in Subsection 6.3.7. Some interesting results are itemized below. The definition of the measures (except φ_R, which is defined in (6.20)) can be found in Chapter 4. Similarly, the static, dynamic, and stationary analysis were defined in Subsection 4.4.

1. The progress rate φ is always nonnegative on ridge functions. This fact contradicts the universal progress law of Rechenberg.

2. The maximum static progress is obtained on the ridge axis.

3. Among all ridge functions, the maximum progress is obtained for the hyperplane given a fixed mutation strength σ.

4. The maximum progress rate for the parabolic ridge is reached as the mutation strength goes to infinity. This limit is nonzero and finite.

5. For the same distance r to the ridge axis, the static progress rate value of the $(1, \lambda)$-ES is larger than the one of the $(\mu/\mu_\mathrm{I}, \lambda)$-ES.

6. The genetic repair hypothesis is observed on the stationary progress rate performance of the $(\mu/\mu_\mathrm{I}, \lambda)$-ES. This algorithm attains in general a smaller value for the stationary distance $R^{(\infty)}$ to the ridge axis than the $(1, \lambda)$-ES or the (μ, λ)-ES, and consequently a larger progress rate.

7. For the parabolic ridge, recombination enables better fulfillment of the short term goal (i.e. minimization of r). As a result of smaller $R^{(\infty)}$, the stationary progress

rate φ is larger, which means that the long term goal is followed better. This is an interesting example for the operation of recombination, and it should be generalizable to other ridge functions with $\alpha > 0$.

8. The progress rate of the $(\mu/\mu_D, \lambda)$-ES depends on how the progress axis is oriented in the search space.

9. An "evolution window" for the mutation strength σ is not observed for $\alpha \leq 2$.

10. A new form of the evolutionary progress principle (EPP) is observed for ridge functions. There is no negative term in the progress rate formula. Instead, the gain term and the loss term are identified in the denominator of the formula.

11. The progress measure φ_R is defined in the search space as the expected decrease in the distance r to the ridge axis in one generation. This measure has been analytically determined for the parabolic ridge. It is used to formalize the measurement of the short term goal (minimize r).

12. The quality gain $\overline{Q}$ is a progress measure in the fitness space. It can be expressed on ridge functions using the progress measures φ and φ_R in the search space.

13. The quality gain cannot be used in general to estimate the progress rate toward the optimum.

14. The quality gain is expected to be equivalent or convertible to the progress rate for the special static case $r = R^{(\infty)}$.

15. The success probability P_{s1} (or equivalently $P_{s\lambda}$) is a measure in the fitness space, and it is defined using the fundamental probability distribution used for the quality gain. For ridge functions, it does not give further information than the quality gain as long as the progress rate is of interest.

16. The success probability P_{s1} (or equivalently $P_{s\lambda}$) is inversely related to the progress rate φ for the static case and a given mutation strength (at least for $1 \leq \alpha \leq 2$): For example, φ attains its maximum value on the ridge axis ($r = 0$), where P_{s1} attains its minimum value. For $r \to \infty$, this relation is reversed.

17. The stationary value of the success probability at the optimum progress rate $\hat{\sigma^*}$ obtained for the parabolic ridge is $P_{s1} = \Phi(-c_{1,\lambda})$ (e.g. $\Phi(-c_{1,10}) \approx 0.062$). It differs considerably from the corresponding value for the sphere model, corridor model, and the hyperplane.

18. The $D^{(\infty)}$ values (see Point 5.3.5.2) play an important role on ridge functions for $\alpha \geq 1$. They emerge as the limit value of the stationary distance $R^{(\infty)}$ for $\sigma \to \infty$. The asymptotic behavior depends on the value of d for finite σ. For $\alpha < 2$, the $R^{(\infty)}$ limit itself will also depend on d.

19. The ES algorithms do not diffuse along the gradient path on ridge functions.

Chapter 7

Experiments

This chapter represents a selection of simulation results obtained. Primarily, they are used to verify the theoretical results from Chapter 6. The asymptotic limits proposed are also justified here. Furthermore, these experiments aim at the comparison of results obtained for convergence measures considering different algorithms as well as different ridge functions. For the special case of elitist strategies, respective experiments investigate the effect of the selection strategy on convergence measures, since no theoretical formulae have been derived for this case.

The experiments are done for the static and stationary cases. Additionally, the mean value dynamics is analyzed for the distance r to the ridge axis. These three analysis methods were briefly described in Subsection 4.4.

The first section (Section 7.1) is dedicated to the analysis of the distance r to the ridge axis. The results obtained are used later in the stationary analysis of φ and P_{s1}, and in the static analysis of $\overline{Q}$. A further reason for this section ordering –which may seem strange– is that the static analysis can treat the distance r as a variable in the experiments. Therefore, this measure was selected for starting the simulations.

Section 7.2 contains the experiments for the progress rate φ. The simulation results for the quality gain $\overline{Q}$ can be found in Section 7.3, and finally the success probability is investigated in Section 7.4. Section 7.5 serves to summarize the results, and concludes the chapter.

The default cases. Each subsection of this chapter contains sufficient information describing the experimental conditions. In order to save space and to avoid repetitions, the default values used in the simulations will be summarized here. The reader can assume the values given here if no further description is provided in the respective subsection.

The theoretical results were derived under the conditions $N \gg \lambda$ (for constant λ or for $\lim_{N\to\infty} \lambda/N \ll 1$ and $N \gg 1$ ($N \to \infty$). The deviations between empirical and theoretical results are generally caused by the violation of this condition. However, the theoretical results will prove themselves to be useful for finite values of N.

The normalizations (6.48) and (6.51) aim at the generalization of simulation results. The simulation results are obtained for $d = 0.01$ and $N = 100$, unless specified otherwise.

In [OBS97], the results obtained using other d values can be found, which gave the same curve as in the $d = 0.01$ case after normalization. Therefore, they are omitted in this chapter.

The simulations are mostly done on the parabolic ridge ($\alpha = 2$) because of its special properties among all ridge functions. The notation "$(1 \overset{+}{,} \lambda)$-ES" (plural) is used to mention both the $(1, \lambda)$-ES and the $(1 + \lambda)$-ES together. The analysis is done using $\lambda = 10$ for simplicity, also for other ES algorithms.

The stationary results are obtained for different cases of $\mathbf{v}$, i.e. for the aligned, diagonal and randomly oriented ones (see Point 6.3.1.6 for formal definitions). The default case is the aligned one. Analytical approximations are principally shown by lines and curves. A horizontal line indicates the limit value of the theoretical curve. The simulation results are shown as "error-bars"; that is, as the mean and its standard error (mean $\pm$ s.e.). The mean value of experimental measurements is displayed by points, and the corresponding standard error as a vertical interval. The standard error is the standard deviation of the mean. It is obtained by dividing the standard deviation of the measured quantity by the square root of the number of measurements taken.

In general, $G = 100\,000$ measurements are made for each single data point in all figures presented in this chapter. An additional 2000-generation period is reserved before starting to collect data for the stationary case in order to guarantee that the transient period is over. To ensure statistical independence, the pseudo-random number generator `ran2` [PTVF92, p.282ff] is initialized to a different random seed for each simulation run. For the $(\mu/\mu_\mathrm{D}, \lambda)$-ES and the (μ, λ)-ES, different copies of the pseudo-random number generator are used for each independent random process (generate offspring, select parent, etc.) and they are initialized independently.

Some simulations were done for investigating the effect of the values of the vector $\mathbf{v}$ (see Equation (3.17)) on the progress rate of ES algorithms, in particular for the $(\mu/\mu_\mathrm{D}, \lambda)$-ES. The aligned and diagonal cases of $\mathbf{v}$ were formalized in (6.59) on Page 93. In the third case, $\mathbf{v}$ is selected randomly for each data point in the simulation series.

As explained in Point 6.3.1.1, the stationary value $R^{(\infty)}$ can be approximated for sufficiently large σ by a simpler formula, which is derived from the $D^{(\infty)}$ value obtained on the sphere model (see Point 5.3.5.2). The appropriateness of this simplification will be shown in this chapter on several experiments. It will be called shortly "the $D^{(\infty)}$-based approximation".

7.1 The distance r to the ridge axis

The distance r to the ridge axis is an important quantity in the analysis of ridge functions. First of all, if its stationary value $R^{(\infty)}$ is available -either from theoretical analysis or from simulation results- one can obtain an approximation for the stationary success probability and progress rate. The necessary formulae were derived in Section 6.2 and Section 6.3, respectively. The $R^{(\infty)}$ value indicates also how well a given ES algorithm satisfies the short term goal (minimization of r, see Subsection 3.3.2).

Secondly, the state equation on r can be used for several purposes. It gives the expected change in one generation. Therefore, it can also be used to estimate the number of generations required to reach the vicinity of $R^{(\infty)}$ for a given $r^{(0)}$. The state equation and the $R^{(\infty)}$ formulae have been derived for the parabolic ridge only. For other ridge functions, the $D^{(\infty)}$-based approximation serves as a lower limit: The ES algorithm attains on the sharp ridge larger $R^{(\infty)}$ values for small d (see Subsection 7.1.2). On ridge function with $\alpha > 2$, larger $R^{(\infty)}$ values are observed for small σ (see Subsection 7.1.3).

This section has six parts. In Subsection 7.1.1, the $R^{(\infty)}$ values of the $(1 \overset{+}{,} 10)$-ES are compared. The comparison is repeated on the sharp ridge in Subsection 7.1.2. Subsection 7.1.3 gives the $R^{(\infty)}$ values for the $(1, 10)$-ES on several ridge functions. The effect of recombination on the $R^{(\infty)}$ value is investigated in Subsection 7.1.4. The last two subsections summarize the static results and their application to the dynamic analysis, obtained using the $(1, 10)$-ES on the parabolic ridge. Subsection 7.1.5 illustrates the progress measure φ_R for various r and σ values. Thereafter, the dynamic analysis shows the applicability of the $r^{(g)}$ mean value dynamics given an initial $r^{(0)}$ value. Furthermore, the time constant ω is calculated numerically for three different $r^{(0)}$ values depending on the product $d\sigma$.

7.1.1 The $R^{(\infty)}$ value for the $(1 \overset{+}{,} \lambda)$-ES

The stationary distance $R^{(\infty)}$ measures how well an ES algorithm has fulfilled the short term goal in the long run. The short term goal was formulated in Subsection 3.3.2 as the minimization of the distance to the ridge axis. The selection scheme used influences the $R^{(\infty)}$ value. Several experiments are carried out for different α values and ES algorithms to measure this value.

In Figure 7.1, the $R^{(\infty)}$ values obtained using the $(1 \overset{+}{,} 10)$-ES are compared. The plus strategy attains smaller $R^{(\infty)}$ values. In other words, it is more successful than the respective comma strategy as long as the short term goal is considered. For small values of the mutation strength, the performance of both algorithms are equally well.

Another aim of this figure is the verification of theoretical results for the $(1, \lambda)$-ES. The $R^{(\infty)}$ formula in (6.166) describes the simulation results successfully. For larger values of σ, it becomes equivalent to the respective $D^{(\infty)}$-based approximation in (6.41). Both formulae underestimate the simulation results by a relative error of 2%, which is considerably small. For small values of σ, $R^{(\infty)}$ is considerably larger than this linear approximation.

If the d value is increased, $R^{(\infty)}$ matches to the linear approximation from a lower value of σ on. This can be easily traced back in the $R^{(\infty)}$ formula (6.166). The squared term in the bracket can be used to express the condition for the validity of the linear $D^{(\infty)}$-based approximation. If $d(N-1)\sigma \gg 2c_{1,\lambda}$, this estimate is applicable. For Figure 7.1, it is expected to hold for $\sigma \gg 3$ (since $d = 0.01$, $N = 100$, $c_{1,10} \approx 1.57$), and it practically holds for $\sigma > 6$. For $d = 0.1$, it is expected to hold if $\sigma \gg 0.3$, and for $d = 0.001$ if $\sigma \gg 30$.

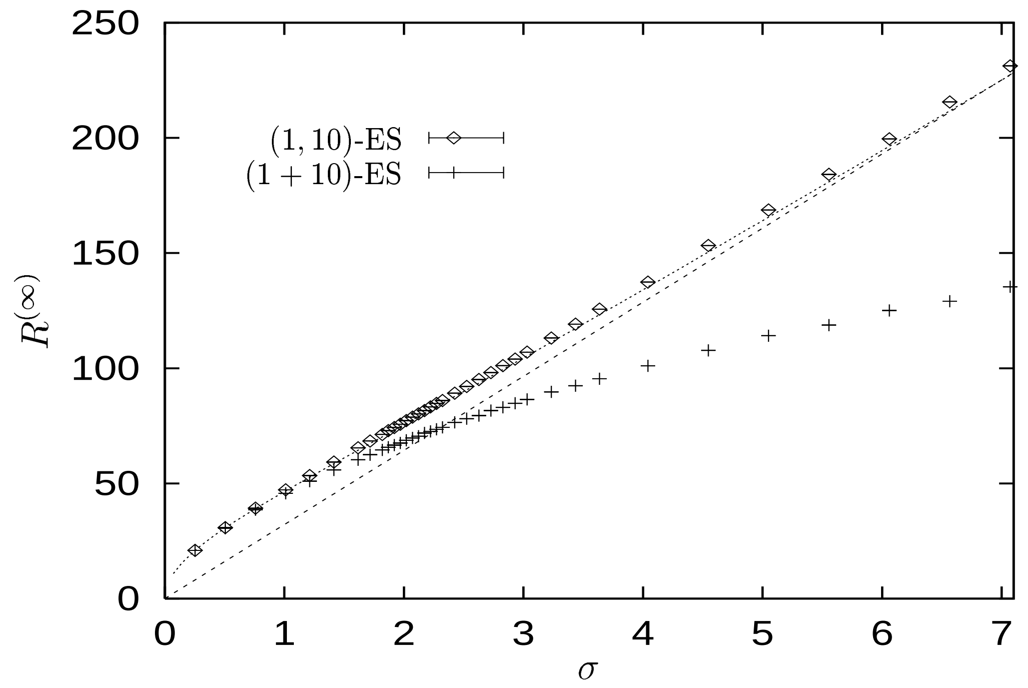

Figure 7.1: The stationary distance $R^{(\infty)}$ versus the mutation strength σ for the $(1 \overset{+}{,} 10)$-ES on the parabolic ridge. The plus strategy provides smaller $R^{(\infty)}$ values. The theoretical $R^{(\infty)}$ curve (6.166) and the corresponding linear $D^{(\infty)}$-based approximation (6.41) are compared with the simulation results for the $(1, 10)$-ES.

7.1.2 The $R^{(\infty)}$ values for the $(1 \overset{+}{,} \lambda)$-ES on the sharp ridge

The performances of the $(1 \overset{+}{,} \lambda)$-ES algorithms in fulfilling the short term goal were compared in Subsection 7.1.1 on the parabolic ridge. A similar comparison is also possible for the sharp ridge. As a primary difference to the parabolic ridge case, one observes

$$R^{(\infty)} \propto \sigma \ . \tag{7.1}$$

In other words, the $R^{(\infty)}$ value increases linearly in σ, and a nonlinear region such as in Figure 7.1 does not exist. As a result, a plot for various values of the mutation strength becomes redundant. It suffices to plot the $R^{(\infty)}$ values for a given σ. The values for other σ can be obtained by a simple multiplication. Therefore, the simulation is done for $\sigma = 1$ and a list of d values.

Figure 7.2 shows that the $R^{(\infty)}$ values are again smaller for the plus case. However, a nonzero $R^{(\infty)}$ value is obtained for both algorithms. In this experiment, the simulation length was chosen as $G = 200\,000$, which gave extremely small standard errors. An analytically derived formula does not exist for $R^{(\infty)}$ values of the $(1, \lambda)$-ES on the sharp ridge. Therefore, these values are compared with the respective $D^{(\infty)}$-based approximation (Equation (6.41), indicated here as a horizontal line). For larger values of d, this result is

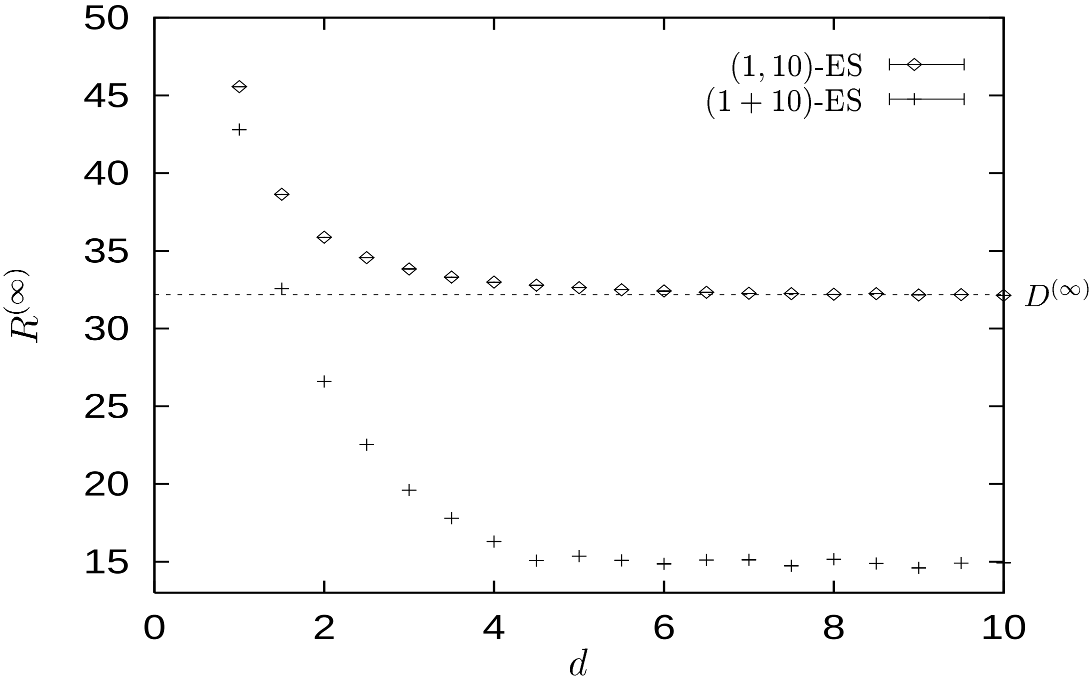

Figure 7.2: The stationary distance $R^{(\infty)}$ versus d for the $(1 \overset{+}{,} 10)$-ES on the sharp ridge. The results are obtained for $\sigma = 1$. The horizontal line is obtained using the $D^{(\infty)}$-based approximation (6.41).

applicable. If one considers the opposite case, $d \to 0$, the $R^{(\infty)}$ values of both ES algorithms go to infinity.

The dependence of the $R^{(\infty)}$ values on d can be explained by looking at the definition of the sharp ridge (3.15). For small values of d, it becomes similar to the hyperplane, and for larger d similar to the sphere model. This is reflected in the fulfillment of the short term goal, as discussed above.

7.1.3 The $R^{(\infty)}$ values for various ridge functions

The $R^{(\infty)}$ values of the $(1, \lambda)$-ES obtained for various ridge functions are compared in Figure 7.3. In order to simplify the comparison, both axes are normalized using the factor $d^{\frac{1}{\alpha-1}}(N-1)$ (see Equation (6.51)). Additional to the results shown in this figure, this normalization scheme gave the same normalized curve for other d values on the parabolic ridge case; for $\alpha = 2$ this is verified by an experiment for $d = 2$.

A nonlinear region can even be observed for the $\alpha = 1.5$ case; however, one observes a different asymptote than for $\alpha \geq 2$. It seems that ridge functions with $0 < \alpha < 2$ have an asymptotic $R^{(\infty)}$ value which cannot be described by using $D^{(\infty)}$. For $1 \leq \alpha < 2$, this asymptotic value is obtained using $D^{(\infty)}$ for sufficiently large d. For $\alpha \geq 2$, this limit is always described using $D^{(\infty)}$, and the value of d determines at which σ value the respective

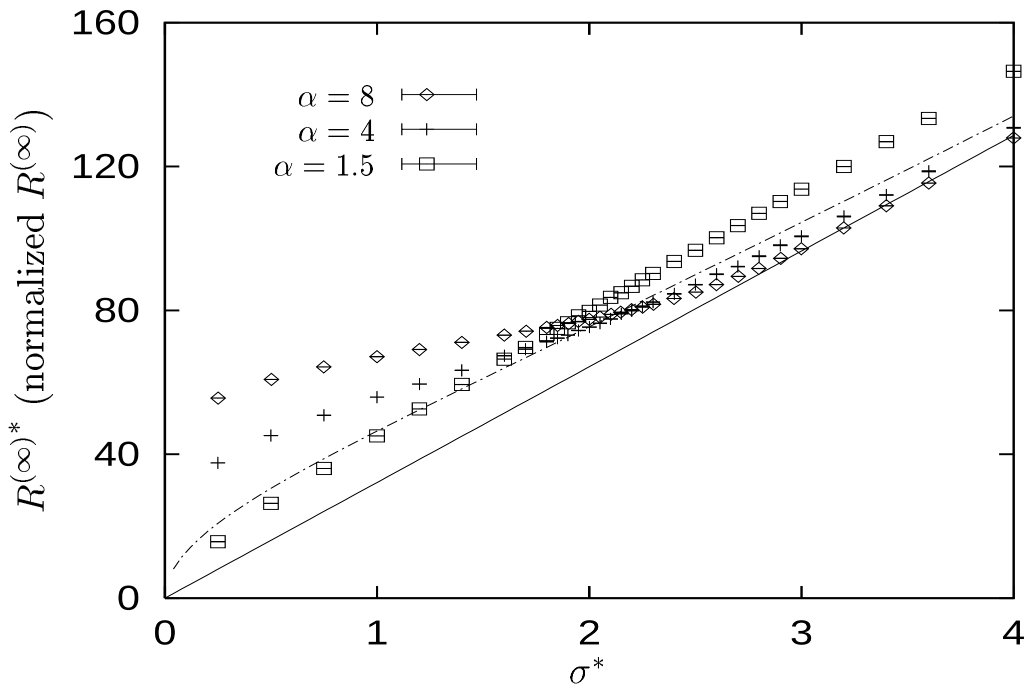

Figure 7.3: The stationary distance $R^{(\infty)}$ versus the mutation strength σ for the $(1, 10)$-ES on various ridge functions for $\alpha \in \{1.5, 4, 8\}$. Both axes are normalized using the factor $d^{\frac{1}{\alpha-1}}(N-1)$. The theoretical $R^{(\infty)}$ curve for the parabolic ridge case (Equation (6.166)) and the corresponding linear $D^{(\infty)}$-based approximation are also displayed.

$R^{(\infty)}$ values can be approximated by $D^{(\infty)}$. In other words, if d gets larger, this limit holds for smaller σ values.

The normalization used clearly expresses where the transition between the nonlinear and linear regions happens for $R^{(\infty)}$. A larger α value results in a larger $R^{(\infty)*}$ in the nonlinear region, and the linear region starts at a smaller σ^* value. For the $\alpha \to \infty$ case, one expects that this transition will be at $\sigma^* \approx 2c_{1,\lambda}$ (please note $2c_{1,10} \approx 3$ in Figure 7.3), which would give $R^{(\infty)*} \approx \sigma^*(N-1)/2c_{1,\lambda} \approx N-1$. This $R^{(\infty)*}$ value corresponds to $R^{(\infty)} = R^{(\infty)*}/d^{\frac{1}{\alpha-1}}(N-1) \approx 1$, independent of the value of d ($d \neq 0$) since $\lim_{\alpha\to\infty} d^{\frac{1}{\alpha-1}} = 1$. For such large α values, the x_0 component dominates the fitness function $F_R(\mathbf{x})$ in (3.13) if $r < 1$. As a result, large $R^{(\infty)*}$ values in the nonlinear region for $\sigma^* < 2c_{1,\lambda}$ are explained. For larger values of σ, the nonlinear part dominates and the distance to the ridge axis increases proportional to σ.

7.1.4 The effect of recombination on the $R^{(\infty)}$ value

This subsection aims at the verification of theoretical $R^{(\infty)}$ formulae for ES algorithms with more than one parent ($\mu > 1$) on the parabolic ridge. The results are shown in

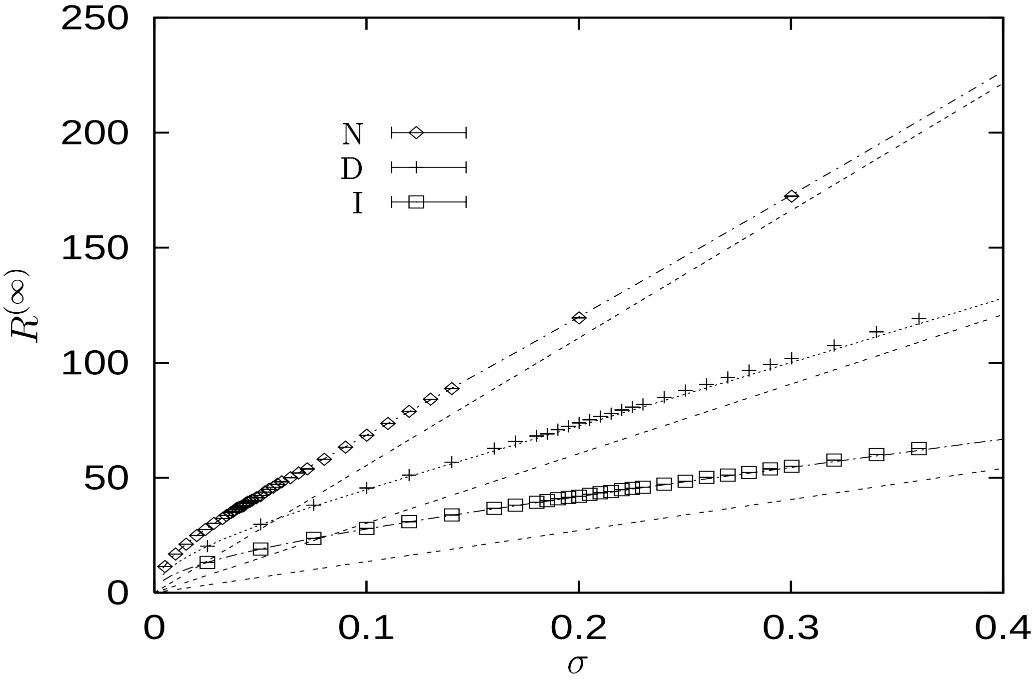

Figure 7.4: The stationary distance $R^{(\infty)}$ versus the mutation strength σ for the $(5, 10)$-ES, the $(5/5_{\mathrm{D}}, 10)$-ES, and the $(5/5_{\mathrm{I}}, 10)$-ES, in decreasing order. The data series for these algorithms are labeled as "N", "D", and "I", respectively. $N = 1000$, $\alpha = 2$. The corresponding theoretical $R^{(\infty)}$ curves are obtained from (6.205), (6.204), and (6.202), respectively. The corresponding linear $D^{(\infty)}$-based approximations are also plotted.

Figure 7.4. For the $(5/5_{\mathrm{D}}, 10)$-ES, the vector $\mathbf{v}$ (see Equation (3.17)) was diagonal in the simulation. If it is aligned with a coordinate axis, different results are obtained for $R^{(\infty)}$. If $R^{(\infty)}$ values are compared quantitatively, one observes that the $(5, 10)$-ES attains the largest $R^{(\infty)}$ values among these three algorithms, and the $(5/5_{\mathrm{I}}, 10)$-ES the smallest. The $R^{(\infty)}$ values for the dominant case can be obtained by using surrogate mutations (see Point 6.3.1.5). For $N = 100$, the asymptotic limits are still valid; however, the $R^{(\infty)}$ values for finite σ are not exactly described by analytical formulae (not shown in the figure).

In Figure 7.4, one can observe that the theoretical predictions accord to simulation results for $N = 1000$. For large σ, they can be approximated by the $D^{(\infty)}$-based approximation. Therefore, as in the $(1, \lambda)$-ES case, the stationary analysis can be carried out simply by substituting $R^{(\infty)}$ into r in the respective formulae for φ, P_{s1}, or $\overline{Q}$.

The calculation of the stationary $\overline{Q}$ is not so simple. The $R^{(\infty)}$ formula used for that purpose contains small approximation errors. Because of these errors, the $\overline{Q}$ formula for the $(1, \lambda)$-ES does not predict the simulation results exactly after the substitution of r by $R^{(\infty)}$. Small approximation errors yield remarkable differences for the stationary analysis of the quality gain $\overline{Q}$. The reason is the strong r-dependence of $\overline{Q}$ values in the vicinity of $R^{(\infty)}$ for large σ values, as will be seen in Subsection 7.3.1.

Comparison of $R^{(\infty)}$ values for the $(1,\lambda)$-ES and $(\mu/\mu_{\rm I},\lambda)$-ES. The theoretical $R^{(\infty)}$ formulae of these two algorithms on the parabolic ridge can be found in (6.166) and (6.202), respectively. It is interesting to investigate for which values of μ the $(\mu/\mu_{\rm I},\lambda)$-ES attains smaller $R^{(\infty)}$ values, since $R^{(\infty)}$ measures how well the short term goal (minimization of r) is fulfilled. For this purpose, these two formulae are compared for large values of the mutation strength ($\sigma \to \infty$). The resulting limits can alternatively be obtained using the idea of Point 6.3.1.1. In other words, these limits are basically equivalent to $D^{(\infty)}$-based approximations on any ridge function for sufficiently large N. Consequently, the $R^{(\infty)}$ value of the $(\mu/\mu_{\rm I},\lambda)$-ES is smaller if

$$\frac{\sigma(N-1)}{2\mu c_{\mu/\mu,\lambda}} < \frac{\sigma(N-1)}{2c_{1,\lambda}} \,, \tag{7.2}$$

$$c_{1,\lambda} < \mu c_{\mu/\mu,\lambda} \,. \tag{7.3}$$

One can make this comparison by using numerical integration or by tables and obtain the largest μ value for which this inequality is valid. Trivially, for $\mu = 1$ one obtains equality. For $\mu = \lambda - 1$, the equality

$$\boxed{c_{1,\lambda} = (\lambda-1)c_{\lambda-1/\lambda-1,\lambda}} \tag{7.4}$$

holds because of the symmetry in the definition of the $c_{\mu/\mu,\lambda}$ integral (5.21). The necessary intermediate step $c_{1,\lambda} = e^{0,1}_{0,\lambda} = e^{1,0}_{1,\lambda}$ can be proven using integration by parts [Bey96c, p. 167], [Bey95b, p. 398]. Consequently, the condition (7.3) holds for $1 < \mu < \lambda - 1$.

As a result of this investigation and generalization of (7.4) one obtains the relation

$$\boxed{\mu c_{\mu/\mu,\lambda} = (\lambda-\mu)c_{\lambda-\mu/\lambda-\mu,\lambda} \,.} \tag{7.5}$$

It holds as a consequence of the symmetry in the $c_{\mu/\mu,\lambda}$ integral. This equality reduces the efforts in the preparation of $c_{\mu/\mu,\lambda}$ tables.

7.1.5 The static progress measure φ_R

The expected progress in r direction was defined in (6.20) as the progress measure φ_R. This measure was used in an alternative derivation of the quality gain $\overline{Q}$ in Subsection 6.1.4. The φ_R value indicates the progress in r direction in one generation. Naturally, this measure is defined in the search space, similar to the progress rate φ.

As one can see in Figure 7.5, the analytical formula (6.177) predicts simulation results well. The results are obtained statically. The stationary case $r \approx R^{(\infty)}$ is also contained in this figure, which yields $\varphi_R = 0$. The simulations are done for the $(1,10)$-ES using the values $\sigma \in \{2, 5, 15\}$ for the mutation strength. The φ_R values are in the interval $-\sigma\sqrt{N-1} \le \varphi_R \le c_{1,10}\sigma$, with $c_{1,10} \approx 1.54$ (see Subsection 5.3.4). The lower limit is sharp for small σ values only. It can also be obtained using (6.174). The upper limit is simply the progress rate limit of the $(1,10)$-ES in the search space. From another point of view, the r value is expected to increase if $r < R^{(\infty)}$, and it will decrease if $r > R^{(\infty)}$ yielding $\varphi_R < 0$ and $\varphi_R > 0$, respectively.

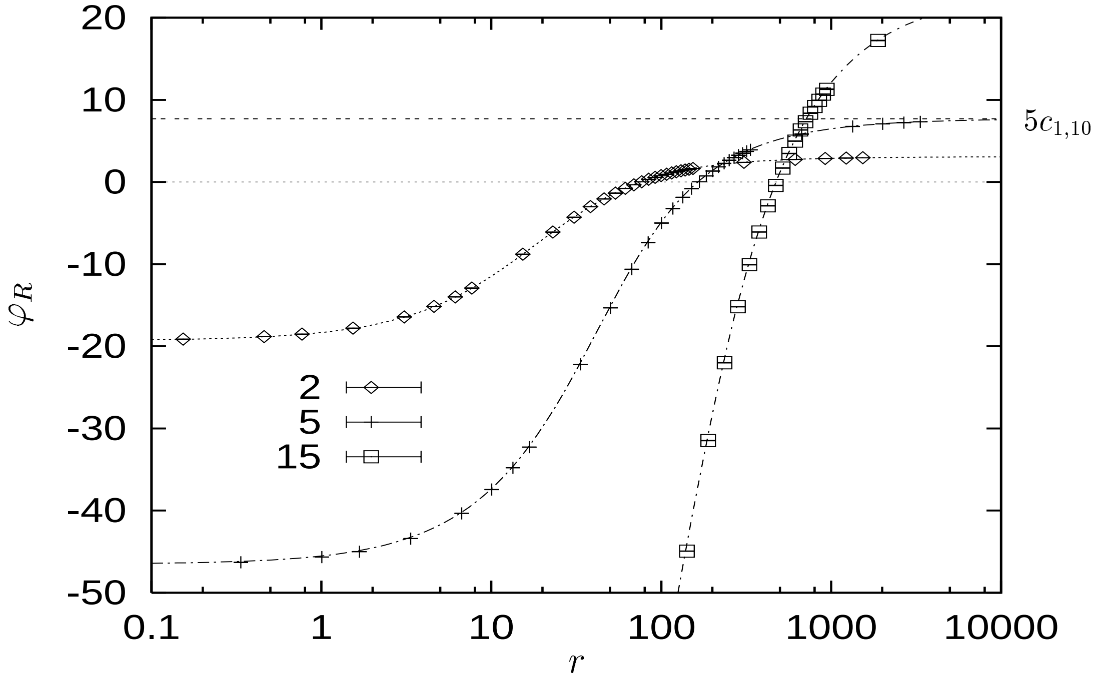

Figure 7.5: The progress measure φ_R versus the distance r to the ridge axis. Static experimental measurements and the theoretical formula (6.177) are plotted for $\sigma \in \{2, 5, 15\}$. The value $\varphi_R = 0$ indicates $r = R^{(\infty)}$. The magnitude of φ_R shows how fast the parent of the $(1, 10)$-ES will approach the stationary distance $R^{(\infty)}$. The upper limit $\varphi_R = c_{1,10}\sigma$ is shown for the case $\sigma = 5$ with a dashed line.

7.1.6 The dynamic analysis and the time constant

The dynamic behavior of the distance r over generations (i.e. of $r^{(g)}$), the so-called mean value dynamics, is another important point to be investigated. For a given $r^{(0)}$ value, the consecutive $r^{(g)}$ values are calculated using (6.168) for the $(1, \lambda)$-ES, and compared with averages obtained from several simulation runs.

The case $r^{(0)} = 0$ is taken as an example. For the mutation strength $\sigma = 1$, the $r^{(g)}$ values are obtained from 100 statistically independent simulation runs. The average of $r^{(g)}$ values is plotted over generations g in Figure 7.6 and compared to the analytical prediction (the solid curve). The accordance of both curves is remarkable. A single run is shown in the same figure using a dotted line in order to indicate that an actual $r^{(g)}$ sequence does not yield a smooth curve.

In the two previous figures, the applicability of the state equation (6.168) was shown for two aspects. The former one, Figure 7.5, has shown that the φ_R formula can be used to estimate the change in r in successive generations. Thereafter, the applicability of this equation to the dynamic analysis was underlined in Figure 7.6. Actually, the most important information obtained from this latter figure is the *time constant* (see Point 6.4.1.7

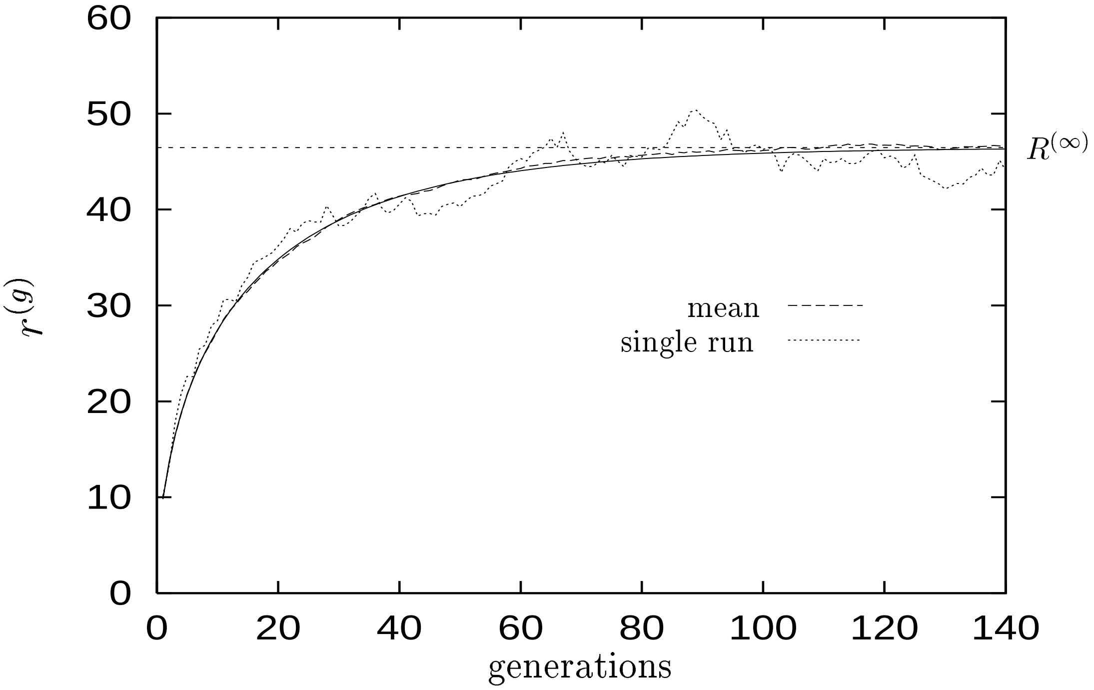

Figure 7.6: The $r^{(g)}$ value dynamics for the $(1, 10)$-ES, $\sigma = 1$, $r^{(0)} = 0$. The horizontal line indicates the $R^{(\infty)}$ value in (6.166). The average over 100 runs is indicated by "mean". A typical single simulation run is also indicated. The theoretical prediction obtained from (6.168) is shown as a solid curve.

for its definition).

One can use the state equation for ${r^{(g+1)}}^2$ successively in order to determine the number of generations necessary to reach an $r^{(g)}$ value in the interval (6.176) for a given $r^{(0)}$. After collecting such data for various σ and d values, one observes that only the product of $d\sigma$ is effective on the time constant. The time constant depends on this product and *not* on individual values of σ and d. Therefore, the time constant is determined for different values of the product $d\sigma$. A theoretical proof for this observation is pending. This hypothesis is tested empirically for $r^{(0)} \in \{10R^{(\infty)}, 2R^{(\infty)}\}$ using a list of d values and $\sigma \in \{0.01, 1\}$. For each $r^{(0)}$ case, the same value for the time constant is obtained for the product $d\sigma$. Only the case $\sigma = 1$ is shown in Figure 7.7 for simplicity. For $r^{(0)} = 0$, this hypothesis is tested for a list of σ values and keeping d constant at several values. The same time constant is obtained for different combinations of σ and d, if their product is the same.

Figure 7.7 displays the numerically obtained time constant values for different $d\sigma$ values starting at three different $r^{(0)}$ values: $10R^{(\infty)}$, $2R^{(\infty)}$, and zero. The last two cases both have the same absolute difference from the stationary value $R^{(\infty)}$; however, in the latter case, the time constant is far smaller. This can be explained by comparing the φ_R values in Figure 7.5 for $r < R^{(\infty)}$ and $r > R^{(\infty)}$. Therefore, it is advisable to initialize $r^{(0)}$ rather smaller than larger if $R^{(\infty)}$ is not known: The resulting transient time will be smaller. The

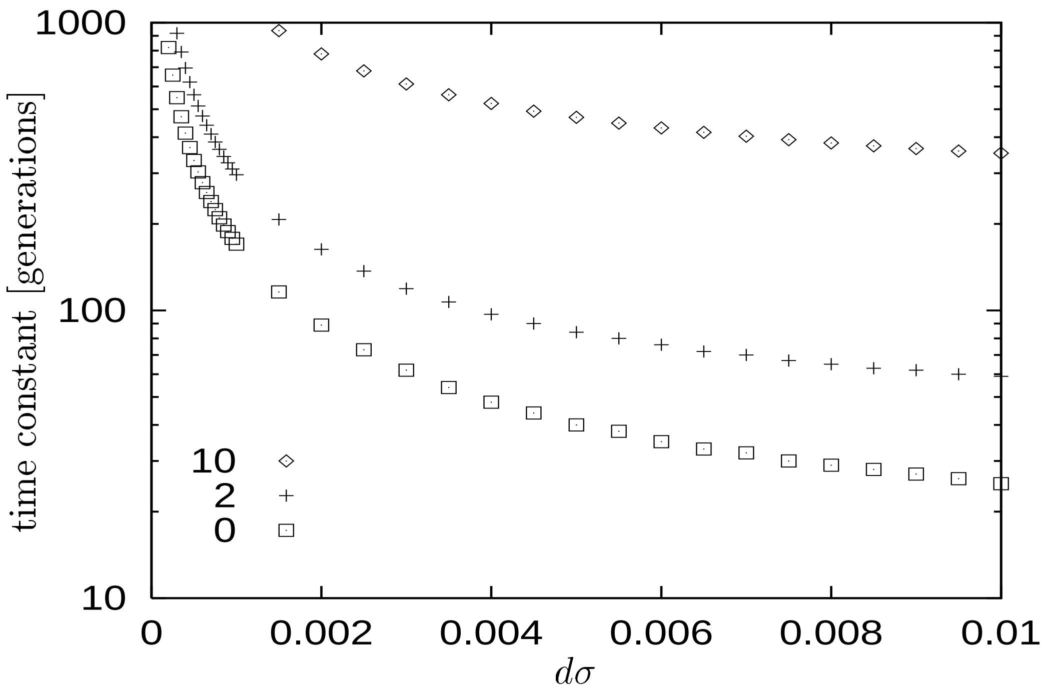

Figure 7.7: The time constant ω versus the products $d\sigma$ for $r^{(0)} \in \{10R^{(\infty)}, 2R^{(\infty)}, 0\}$. For larger values of the product $d\sigma$, the interval given in (6.176) is reached faster. These numerical results are obtained by successively applying the state equation (6.168) for the $(1, \lambda)$-ES.

first case $r^{(0)} = 10R^{(\infty)}$ is depicted in order to show that even in this case the time constant is not very large.

The calculation of the time constant for very large $r^{(0)}$ is relatively simple. For $r^{(0)} \gg 10R^{(\infty)}$, the distance to the ridge axis decreases by $\varphi_R = c_{1,\lambda}\sigma$ per generation. This decrease becomes slower as the $r^{(g)}$ values close to $10R^{(\infty)}$ are attained. The number of generations to attain $10R^{(\infty)}$ can simply be obtained, which is approximately $(r^{(0)} - 10R^{(\infty)})/c_{1,\lambda}\sigma$. Thereafter, the time constant for $r^{(0)} = 10R^{(\infty)}$ must be added to this partial result. Furthermore, the *lower limit* for the time constant can be stated as (please note that $R^{(\infty)} \propto \sigma$)

$$\text{time constant} \quad \omega > |r^{(0)} - R^{(\infty)}|/c_{1,\lambda}\sigma \ . \tag{7.6}$$

For instance, one obtains for the three cases $r^{(0)} \in \{10R^{(\infty)}, 2R^{(\infty)}, 0\}$ the values 20.4, 20.4, and 183.6 respectively (for $\sigma \gg 1$). This lower limit is not sharp. Since $|\varphi_R| \gg c_{1,\lambda}\sigma$ for $r \approx 0$ (see e.g. Equation (6.171)), this lower limit prediction is not necessarily applicable for $r^{(0)} < R^{(\infty)}$.

Figure 7.7 displays the time constant values for $d\sigma \leq 0.01$. The state equation (6.168) is further used to predict the time constant for larger values of the product $d\sigma$. For larger values of $d\sigma$, the theoretical approximation of the time constant decreases further. It becomes $\omega_{10} = 251$ for the $r^{(0)} = 10R^{(\infty)}$ case; the number of generations obtained for the latter two cases were $\omega_2 = 43$ and $\omega_0 = 17$, respectively [OBS97, p.59-62]. In the simulations of this work, the values $d\sigma \geq 0.025$ are relevant since the minimum values used were $\sigma = 0.25$ and $d = 0.01$ for the parabolic ridge. Therefore, the number of generations (i.e. 2000 used) for the transition to the stationary state suffices.

7.2 The progress rate φ

This section summarizes the simulation results for the progress rate φ. The progress measure φ (4.6) is defined in the search space. Its values naturally depend on the ridge function analyzed and the ES algorithm used.

This section has ten subsections. The first eight are devoted to the stationary analysis, the last two address the static case. The parabolic ridge is considered exclusively in first four subsections and the ninth one, and the sharp ridge in the fifth one. Other subsections are on other ridge functions, or they serve for comparisons made on different ridge functions.

In Subsection 7.2.1, the N-dependence of the asymptotic ($N \to \infty$) progress rate formula for the $(1, \lambda)$-ES is investigated on the parabolic ridge. Moreover, the performance of $(1 \overset{+}{,} \lambda)$-ES are compared for $\lambda = 10$. The empirical results for stationary progress rate of the (μ, λ)-ES on the parabolic ridge are compared with the theoretical predictions (Subsection 7.2.2). The same comparison is done for the $(\mu/\mu_\mathrm{I}, \lambda)$-ES in Subsection 7.2.3, and for the $(\mu/\mu_\mathrm{D}, \lambda)$-ES in Subsection 7.2.4. Subsection 7.2.5 investigates the progress rate formula of the $(1, \lambda)$-ES on the sharp ridge, and compares them with the simulations of the $(1 + \lambda)$-ES.

The performance of the $(\mu/\mu_\mathrm{I}, \lambda)$-ES on the ridge function with $\alpha = 5$ is analyzed using the simple local model in Subsection 7.2.6. The analysis for the $(1, \lambda)$-ES case can also be found there. Subsection 7.2.7 shows the rotation-dependence of the performance of the $(\mu/\mu_\mathrm{D}, \lambda)$-ES. The results on the ridge functions with $\alpha = 4$ and $\alpha = 0$ are used for this purpose. A comparison of progress rate curves of the $(1, \lambda)$-ES on different ridge functions can be found in Subsection 7.2.8.

Subsection 7.2.9 shows how different the progress rate φ and quality gain $\overline{Q}$ can be if measurements are taken statically at $r = 0$ (see also the Subsection 7.3.2 for a static comparison for various r values). The results of the $(1, \lambda)$-ES on the parabolic ridge are used for this purpose. Its progress rate and quality gain values are additionally contrasted to the ones of the $(1 + \lambda)$-ES. The last subsection (Subsection 7.2.10) is devoted to static φ results of the $(1, \lambda)$-ES on the sharp ridge, parabolic ridge, and the ridge with $\alpha = 8$.

7.2.1 The N dependence

The normalized progress rate formula of the $(1, \lambda)$-ES in (6.89) was derived for the asymptotic ($N \to \infty$) limit case. Formula (6.48) was used in the normalization. This progress rate formula gives satisfactory results for finite N. The N-dependence of simulation results is exemplified in Figure 7.8 for the $(1, 10)$-ES. The asymptotic limit $c_{1,\lambda}^2$ is displayed in this

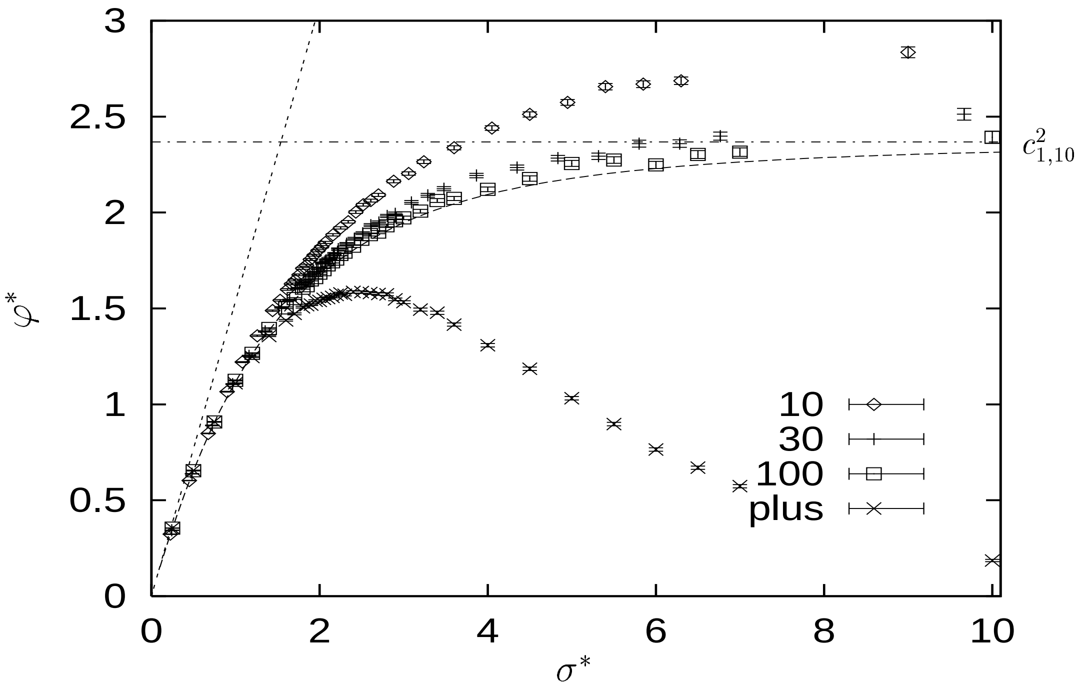

Figure 7.8: The progress rate φ on the parabolic ridge versus the mutation strength σ. Both axes are normalized using (6.48). The simulation results for $N \in \{10, 30, 100\}$ are compared to the theoretical formula in (6.89). Additional to these results for the $(1, 10)$-ES, the simulation results for the $(1 + 10)$-ES are also shown ($N = 100$ case). The horizontal line $c^2_{1,10}$ and the limit $\varphi^* \leq c_{1,10}\sigma^*$ are also shown.

figure as a horizontal line. As N is increased, the simulation results are described more accurately by the theoretical formula (6.89).

The φ formula (6.45) derived by using the local model gives accurate results for $N = 100$ as long as the $R^{(\infty)}$ values obtained from simulations or from the theoretical $R^{(\infty)}$ formula (6.166) are used. For the cases $N = 10$ and $N = 30$, it does not give accurate results. The theoretical results obtained from the local model are omitted in the figure since they do not differ much from (6.89).

The simulation results for φ^* obtained using the $(1 + 10)$-ES are also depicted in the figure. They are obtained for $N = 100$. After comparing them with the results obtained for the $(1, 10)$-ES under the same conditions, one observes significant differences. For $\sigma^* < 2$, both the $(1 + 10)$-ES and the $(1, 10)$-ES attain almost the same progress rate performance. The progress rate of the $(1 + 10)$-ES is slightly larger for σ^* values where both algorithms attain progress rate values as large as the hyperplane. However, for larger values of σ^*, the φ^* value of comma strategy increases further to an asymptote, whereas the values for plus strategy gradually decrease down to zero. The peak performance $\hat{\varphi}^*$ of the comma strategy is larger.

Another set of simulations was done for keeping N constant (e.g. $N = 100$) and increasing λ (not shown here). The simulations for $\lambda = 100$ and $\lambda = 500$ gave smaller deviations from the theoretical formula than in the low N case. These deviations have almost vanished for larger N (e.g. at $N = 1000$ for $\lambda = 100$). The elitist $(1+\lambda)$-ES gave smaller progress rates and peak performances than the $(1,\lambda)$-ES under the same conditions (not shown here). The detailed results can be found in [OBS98a, p. 21ff].

The quality gain $\overline{Q}$ is also measured in these simulations. It gave the same results as φ in the stationary case since the linear component of the fitness function does not have a factor. However, the standard error of the $\overline{Q}$ measurements is much larger, since the variables x_i in (3.11) are squared for the parabolic ridge and the fluctuations of them are reflected in the fitness values. Moreover, this standard error increases in σ much faster than the one of φ because of the same reason. The stationary $\overline{Q}$ measurements and comparative figures of standard errors for φ and $\overline{Q}$ are omitted here. Additional explanatory figures can be found in [OBS97, p. 43].

7.2.2 Increasing the number of parents μ

The normalized progress rate φ^* of the (μ,λ)-ES on the parabolic ridge will be compared for different μ values. The values $\lambda = 10$ and $\mu \in \{2, 5, 9\}$ are used in simulations. The results are in accordance to the formula (6.125). The asymptotic $(\sigma \to \infty)$ φ^* values are also predicted correctly.

In this simulation, one observes that the rotation of the vector $\mathbf{v}$ in (3.17) does not influence the progress rate of the (μ,λ)-ES. One also observes that the simulation results for $N = 100$ and $N = 1000$ become equivalent after normalization. The conclusion obtained from these experiments with different μ values is that the $(1,\lambda)$-ES attains the largest progress rate among the (μ,λ)-ES. An increase in the number of parents for a given λ causes a decrease in the progress rate. The results for the $(1,\lambda)$-ES case can be found in Subsection 7.2.1.

7.2.3 The effect of intermediate recombination

The progress rate value for the $(\mu/\mu_\mathrm{I},\lambda)$-ES is expected to surpass the one of the $(1,\lambda)$-ES and the (μ,λ)-ES. The formulae derived in Section 6.3.3 support this expectation. The prediction power of these formulae will be validated here for different values of μ.

The choice of the optimal μ value for a given number of offspring is investigated in Figure 7.10 for the $(\mu/\mu_\mathrm{I}, 10)$-ES. An important result for larger values of μ was referred in Subsection 6.3.6. In Figure 7.10 ($\lambda = 10$), one observes that $\mu = 3$ gives the largest progress rate value, and $\mu = 9$ the smallest one. The simulation results of the $(\mu/\mu_\mathrm{I}, 10)$-ES can be compared directly with the ones of the $(\mu, 10)$-ES in Subsection 7.2.2 for $\mu = 5$ and $\mu = 9$. The effect of recombination in increasing the progress rate performance is remarkable. One also observes that the choice of $\mathbf{v}$ does not affect the φ^* values.

The progress rate curve for the $(4/4_\mathrm{I}, 10)$-ES and its asymptotic limit $4c^2_{4/4,10}$ are not shown. This algorithm gives a progress rate performance between the cases $\mu = 2$ and $\mu = 5$ of the $(\mu/\mu_\mathrm{I}, 10)$-ES. One can conclude that the recombination operator increases

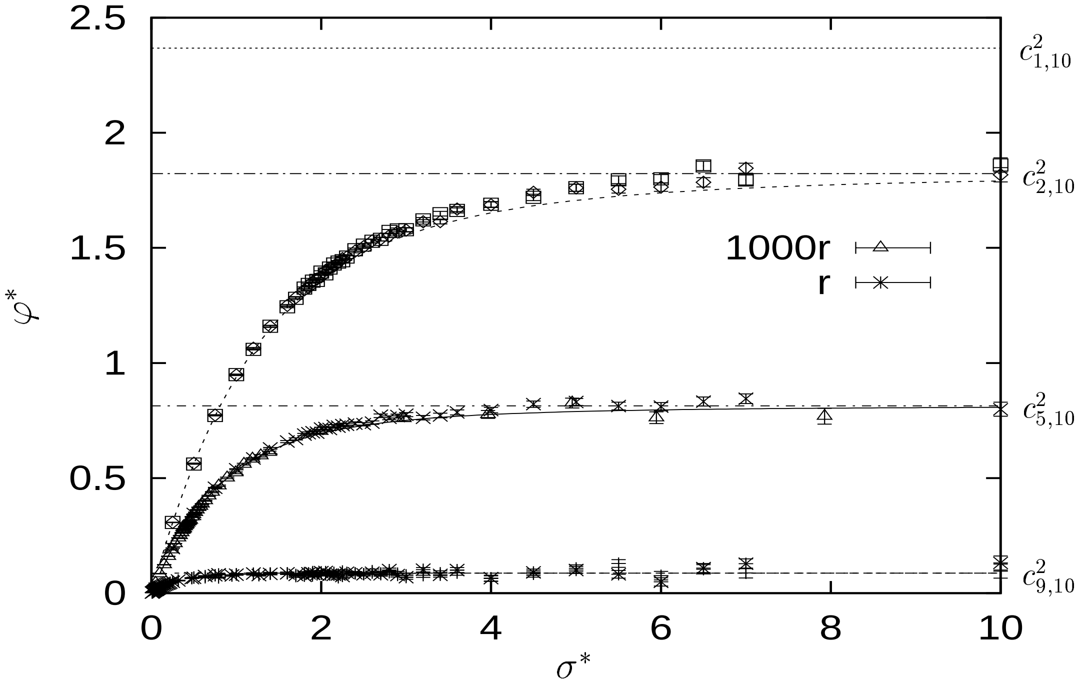

Figure 7.9: The progress rate φ versus the mutation strength σ for the $(\mu, 10)$-ES on the parabolic ridge $(\alpha = 2)$. The simulation results are obtained for $\mu \in \{2, 5, 9\}$. The normalization (6.48) was used for both axes. Four horizontal lines indicate the asymptotic limits of ES algorithms in descending order $c^2_{1,10} > c^2_{2,10} > c^2_{5,10} > c^2_{9,10}$ (The topmost one is for the $(1, 10)$-ES, the lowest for the $(9, 10)$-ES). The notation "`r`" in the legend denotes that the vector $\mathbf{v}$ in (3.17) was chosen randomly and "`1000`" means $N = 1000$ was used in the simulation. The progress rate formula (6.125) is also indicated in the figure.

the performance of the ES algorithm for a large interval of parents: For $\mu \leq 5$, the $(\mu/\mu_I, \lambda)$-ES surpasses the $(1, \lambda)$-ES on the parabolic ridge. As an important remark, the $(5/5_I, 10)$-ES yields a larger φ^* than the $(1, 10)$-ES although it works with relatively large selection ratio.

In Figure 7.10, the theoretical progress rate curve accords to simulation results for $N = 1000$ (or $N \geq 1000$) and not for $N = 100$, which has successfully been done for the $(1, 10)$-ES in Figure 7.8. This can clearly be seen for the $\mu = 3$ case. This fact should be considered in the utilization of asymptotic $(N \to \infty)$ formulae *with* recombination. In other words, the formula derived under the condition $(N \to \infty)$ can be used starting from a larger value of N. Otherwise, the N-dependent progress rate formulae should be used, which has not been derived yet. Fortunately, the asymptotic values are attained even in this case $(N = 100)$, although at larger σ^* values.

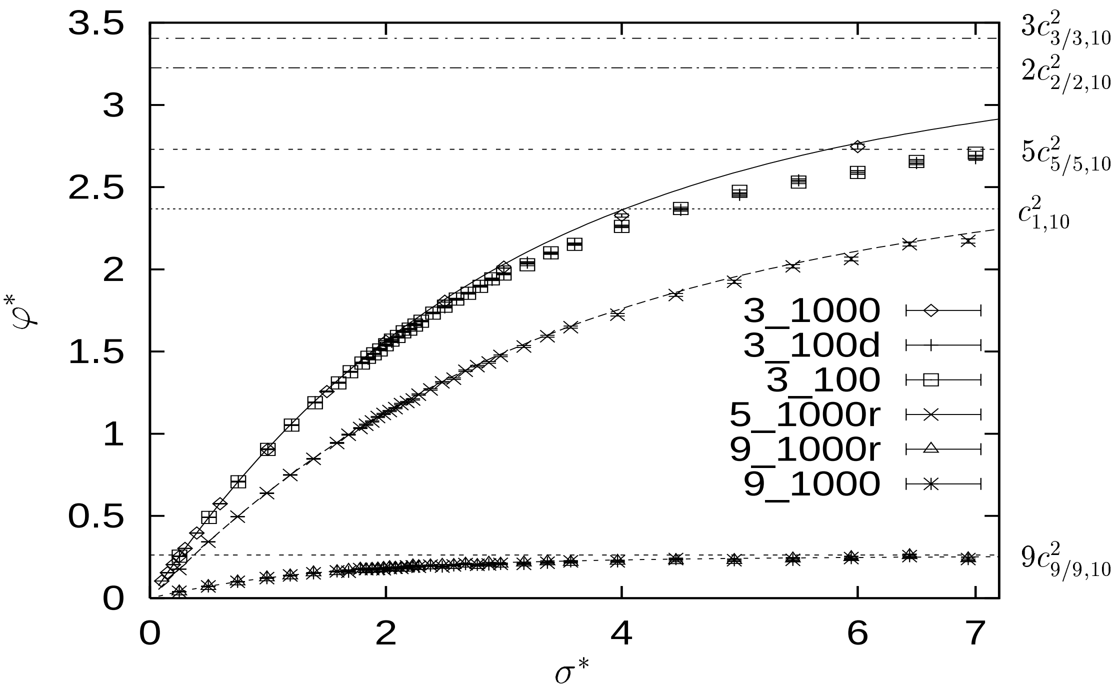

Figure 7.10: The progress rate φ versus the mutation strength σ for the $(\mu/\mu_{\mathrm{I}}, 10)$-ES on the parabolic ridge ($\alpha = 2$), $\mu \in \{3, 5, 9\}$. The normalization (6.48) was used for both axes. Five asymptotic limits drawn are $3c^2_{3/3,10} > 2c^2_{2/2,10} > 5c^2_{5/5,10} > c^2_{1,10} > 9c^2_{9/9,10}$. The φ^* values are predicted correctly by the formula (6.113) for $N = 1000$. In the legend, the letters "d" and "r" indicate that the vector $\mathbf{v}$ was diagonal or randomly chosen, respectively. The value of μ and N are indicated for each simulation run, too.

7.2.4 The effect of dominant recombination

The progress rate φ^* of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES was estimated in (6.119). The aim of this subsection is to find out whether this formula provides useful predictions and to state the necessary conditions for that. It was conjectured that this formula is only valid for diagonal $\mathbf{v}$ (see Equation (6.58)). Therefore, one should also make an experiment for the other extreme, i.e. if vector $\mathbf{v}$ is aligned (see Equation (6.57)).

The effect of orientation of $\mathbf{v}$ is tested on the $(2/2_{\mathrm{D}}, 10)$-ES. The first comparison is for $N = 100$. The progress rate increases faster for the diagonal case: Please see (6.59) for understanding larger φ^* values of the diagonal $\mathbf{v}$ case at low σ^* values. However, the aligned case attains similar values for larger σ^*. Moreover, they both attain the theoretical limit $2c^2_{2/2,10}$ for $\sigma^* \to \infty$ (not shown here). If one repeats the experiment for the diagonal case with $N = 1000$, one observes that the theoretical formula for φ^* holds. This is more or less the case for the other experiment with $\mu = 5$, although the deviations are larger.

In summary, the theoretical formula for φ^* of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES holds for the parabolic ridge if N is sufficiently large. For smaller N, the simulation results for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES

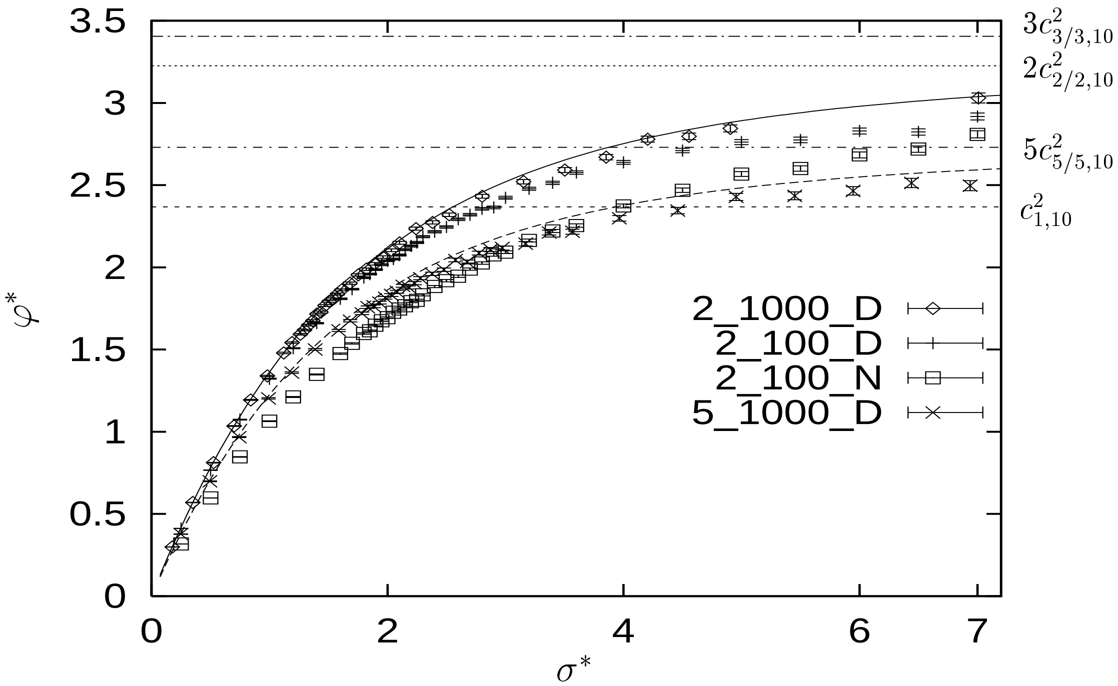

Figure 7.11: The progress rate φ versus the mutation strength σ for the $(\mu/\mu_{\mathrm{D}}, 10)$-ES on the parabolic ridge ($\alpha = 2$), for $\mu = 2$ and $\mu = 5$, using (6.48) for normalization. Four asymptotic limits are displayed $3c^2_{3/3,10} > 2c^2_{2/2,10} > 5c^2_{5/5,10} > c^2_{1,10}$. The formula (6.119) for φ^* is plotted for $\mu \in \{2, 5\}$. The N values (100 and 1000) and the μ values are also indicated in the legend, as well as whether $\mathbf{v}$ is aligned ("N") or diagonal ("D").

were smaller than the values given by this formula. In contrast, the normalized progress rate values of the $(1, \lambda)$-ES were larger than the analytical formula for smaller N (see Subsection 7.2.1).

Furthermore, Figure 7.11 contains the asymptotic limits for some $(\mu/\mu_{\mathrm{D}}, 10)$-ES algorithms ($\mu \in \{2, 3, 5\}$) and for the $(1, 10)$-ES on the parabolic ridge. The simulation results and the predictions are not shown for $\mu = 3$ although a good accordance is also observed for this case. As one can see, the progress rate performance of the $(1, 10)$-ES is surpassed by the $(\mu/\mu_{\mathrm{D}}, 10)$-ES for $2 \leq \mu \leq 5$ (see also Subsection 7.2.3 for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES). Furthermore, the surrogate mutation model in Point 6.3.1.5 holds to convert the results of dominant and intermediate recombination to each other for sufficiently large N ($N \geq 1000$).

7.2.5 The $(1 \overset{+}{,} \lambda)$-ES on the sharp ridge

The progress rate of the sharp ridge in (6.92) was obtained as a special case of the general ridge formula for $\alpha = 1$. Alternatively, the local model yields the formula (6.52), which was expected to give similar results for the stationary case. In the former formula, the $R^{(\infty)}$ values obtained from simulation results are inserted for r. This formula gives approximately

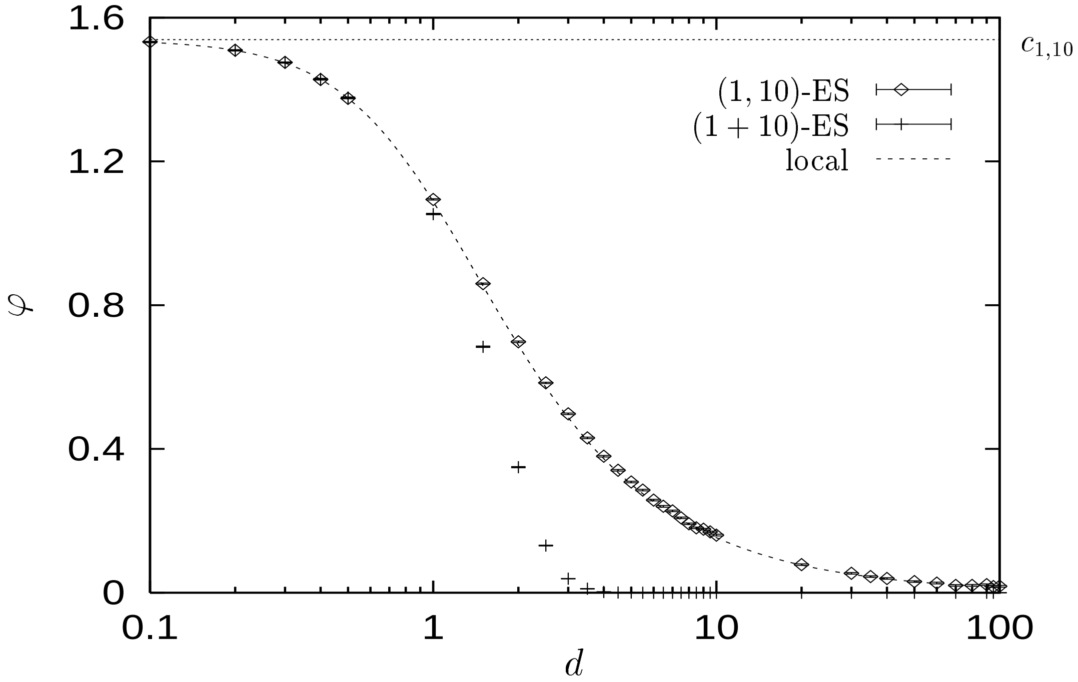

Figure 7.12: The progress rate φ versus d for the $(1 \overset{+}{,} 10)$-ES on the sharp ridge, $\sigma = 1$. The theoretical formula (6.52) using the simple local model is plotted. The upper limit $c_{1,10}$ is obtained for $d \lessapprox 0.1$. The stationary progress rate of the $(1 + 10)$-ES appears small compared to the $(1, 10)$-ES for $d > 4$.

the same results as the latter one in the stationary case. Therefore, it is omitted in Figure 7.12.

The maximum φ value of the $(1, 10)$-ES for $d \to 0$ is obtained as $c_{1,10} \approx 1.54$ (see Subsection 5.3.4 for $c_{1,10}$). The value $\sigma = 1$ is used for the mutation strength in simulations. The results for other σ can be obtained by a simple multiplication since the progress rate is proportional to the mutation strength ($\varphi \propto \sigma$) for the $(1 \overset{+}{,} \lambda)$-ES. This proportionality also holds for $\mu > 1$. For example, the curves for the $(\mu/\mu_I, \lambda)$-ES and $\sigma = 1$ are similar to the $(1, \lambda)$-ES case, with the maximal value $c_{\mu/\mu,\lambda}$ instead. Therefore, the corresponding figures are omitted.

Furthermore, Figure 7.12 serves as a comparison of the $(1, \lambda)$-ES and the $(1 + \lambda)$-ES on the sharp ridge. For $d > 4$, the stationary progress rate of the $(1 + 10)$-ES is small compared to the $(1, 10)$-ES. The plus strategy attains poorer performance for $d \gtrapprox 1$.

7.2.6 The $(\mu/\mu_I, \lambda)$-ES on the ridge function with $\alpha = 5$

The effect of intermediate recombination on the progress rate φ is investigated in this section on the ridge function with $\alpha = 5$. Recombination is expected to increase the progress rate in the stationary case, since the theoretical analysis predicted lower $R^{(\infty)}$

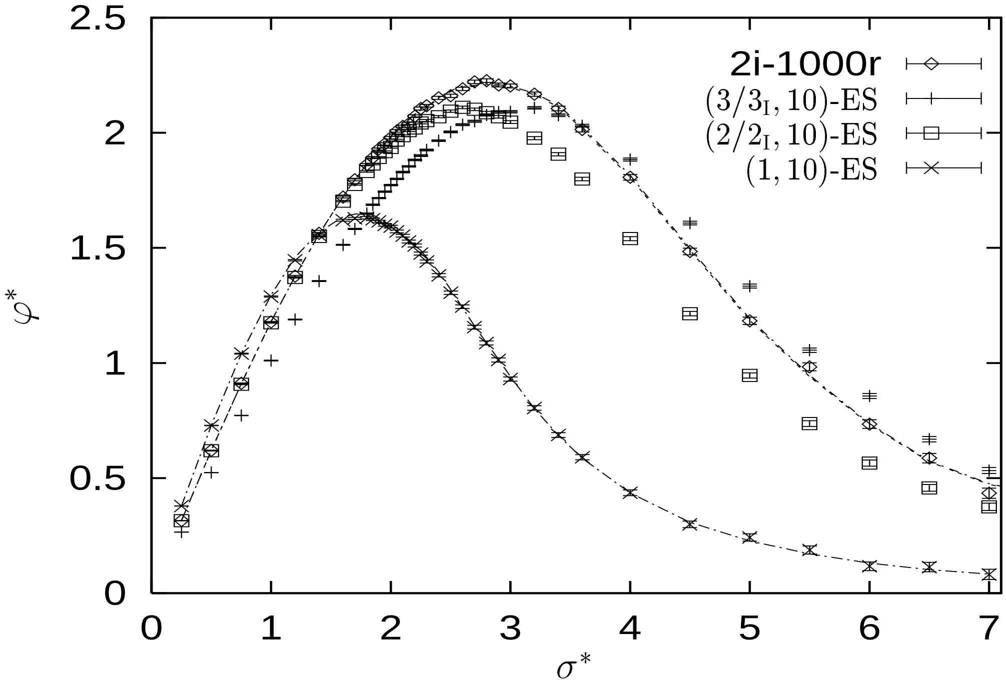

Figure 7.13: The progress rate φ versus the mutation strength σ of the $(\mu/\mu_{\mathrm{I}}, 10)$-ES on the ridge function with $\alpha = 5$ using the normalization (6.51). The cases $\mu = 2$ and $\mu = 3$ are shown. The former experiment is repeated for $N = 1000$ and randomly chosen vectors $\mathbf{v}$ (labeled as "`2i-1000r`"). The normalized versions of (6.108) and (7.7) do not differ for this case. The results for the $(1, 10)$-ES are depicted for comparison along with the theoretical formula (6.45).

values than the one obtained for the $(1, \lambda)$-ES. The second aim of this subsection is the verification of the theoretical formula obtained for the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES on the general ridge case.

Figure 7.13 shows normalized progress rate curves for the $(2/2_{\mathrm{I}}, 10)$-ES and the $(3/3_{\mathrm{I}}, 10)$-ES ($N = 100$, $\alpha = 5$). Equation (6.51) is used for the normalization. The definition of the general ridge function can be found in (3.17). The maximum progress rate φ^* obtained in both cases are almost the same, although the corresponding σ^* values differ. The theoretical formula obtained using the local model in Point 6.3.1.3

$$\varphi \approx \frac{c_{\mu/\mu,\lambda}\sigma}{\sqrt{1 + \left(d\alpha r^{\alpha-1}\right)^2}} \tag{7.7}$$

overestimates these experimental results. Therefore, the curves from this formula are not drawn for these two cases with $N = 100$. The normalization (6.51) and $R^{(\infty)}$ values from simulation results are used in the comparison.

The simulation for $\mu = 2$ is repeated for $N = 1000$. The vector $\mathbf{v}$ is randomly directed in the search space to see whether the progress rate values depend on its direction. It is sampled anew for each data point. As expected, (7.7) obtained using the simple local model predicts the simulation results well. The static formula (6.108) is also shown in the figure with using $R^{(\infty)}$ values obtained from simulations. It gives almost the same curve as the formula obtained using the simple local model. Additionally, one observes that the simulation results do not depend on how $\mathbf{v}$ is directed.

The curve for the $(1, 10)$-ES is also displayed in the figure. The formula of the local model matches in this case even for $N = 100$. Since no recombination is used in the $(1, \lambda)$-ES algorithm, this simple asymptotic $(N \to \infty)$ formula holds for smaller values of N. An analytic explanation for this observation is pending: The asymptotic $(N \to \infty)$ formula of the $(\mu/\mu_{\mathrm{I}}, 10)$-ES seems to be valid for $N \gtrapprox 1000$. The peak performances $\hat{\varphi}^*$ of the $(\mu/\mu_{\mathrm{I}}, 10)$-ES with $\mu = 2$ and $\mu = 3$ surpass the one of the $(1, 10)$-ES.

Two interesting observations should be added at the end: Firstly, the $(1, 10)$-ES attains higher φ^* values for small σ^* ($\sigma^* \lessapprox 1$) since its $R^{(\infty)}$ value does not differ much from the one of the $(\mu/\mu_{\mathrm{I}}, 10)$-ES. This was also observed on the sphere model [Bey96c, p. 212ff]. However, the algorithms with intermediate recombination give much better results because of small $R^{(\infty)}$ values for $\sigma^* \gtrapprox 4$. This becomes clear after comparing (7.7) and (6.45): The $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES must have much smaller $R^{(\infty)}$ values to surpass the $(1, \lambda)$-ES. Additionally, one also has to consider that the $c_{\mu/\mu,\lambda}$ values are smaller than the $c_{1,\lambda}$ value for $\mu > 1$ (see e.g. Table 5.1 on Page 62). Nevertheless, the progress rate of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES is larger.

These results should be considered as an evidence for the genetic repair hypothesis (Subsection 5.2.7) on the distance r to the ridge axis, i.e. on the short term goal. According to this hypothesis, a reduction in the (harmful) mutation components perpendicular to the progress direction is expected if the recombination operator is used. This reduction caused on the sphere model a considerable decrease in the loss term, and consequently an increase in the progress rate value (see the evolutionary progress principle, Subsection 5.2.6). For ridge functions, such a reduction is observed in the stationary value of the orthogonal components (i.e. smaller $R^{(\infty)}$). This reduction caused a smaller denominator in the progress rate formula and consequently a larger φ. In other words, the ES algorithms with the recombination operator fulfill the short term goal better, and as a consequence, they are more successful in fulfilling the long term goal. The definitions of the long term goal and the short term goal can be found in Subsection 3.3.2.

7.2.7 The $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES on ridge functions with $\alpha = 4$ and $\alpha = 0$

This subsection summarizes important results obtained for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES on the general ridge function (3.17). The experiments are done for the ridge function with $\alpha = 4$ and $\alpha = 0$. The dependence of its performance on the vector $\mathbf{v}$ is shown as well as the appropriateness of the surrogate mutations' estimate. The simple local model gives accurate results for the progress rate.

Figure 7.14 shows the dependence of the progress rate of the $(9/9_{\mathrm{D}}, 10)$-ES on the direction of $\mathbf{v}$. The progress rate limit $\varphi^* = \sqrt{9} c_{9/9,10} \sigma^* \approx 0.51\sigma^*$ obtained from (6.59) is also shown in the figure (straight double dashed line). The lowest performance is obtained if $\mathbf{v}$ is aligned with the ridge axis (case "N"). The results for diagonal $\mathbf{v}$ (case "D") and

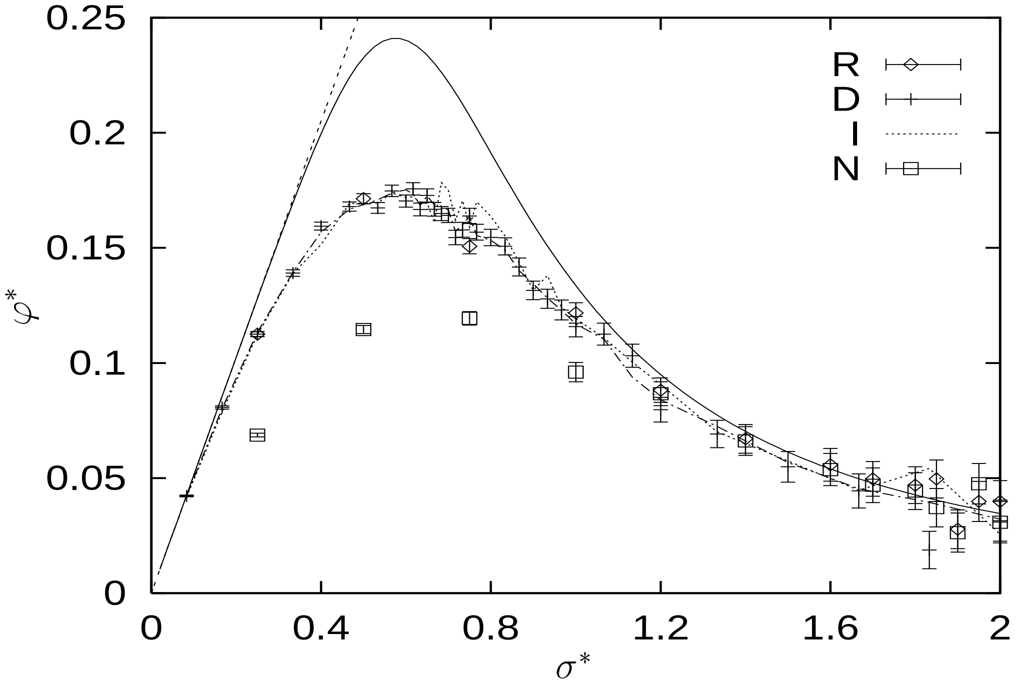

Figure 7.14: The progress rate φ versus the mutation strength σ of the $(9/9_{\mathrm{D}}, 10)$-ES on the ridge function with $\alpha = 4$, using the normalization (6.51), $N = 1000$. Three cases are considered: With an aligned $\mathbf{v}$ ("N", Equation (6.57)), with diagonal $\mathbf{v}$ ("D", Equation (6.58)), and with randomly chosen $\mathbf{v}$ vectors ("R"). The results for the $(9/9_{\mathrm{I}}, 10)$-ES are shown using dotted lines (after the appropriate transformation for surrogate mutations) as the case "I". The solid curve shows the formula in (7.10). Using the $R^{(\infty)}$ values from simulations in (7.9), one obtains a good accordance (the "$- \cdot -$" curve).

randomly selected $\mathbf{v}$ (case "R") do not differ much from each other. In the latter one, $\mathbf{v}$ is selected anew for each data point of the simulation series. This data series differs form the theoretical prediction more than the former one, which should be regarded as an evidence for the influence of the orientation of $\mathbf{v}$ on the progress rate. The former case accords to the estimate obtained using the local model, (6.45), (6.59)

$$\varphi \approx \frac{\sqrt{\mu} c_{\mu/\mu,\lambda} \sigma}{\sqrt{1 + (d\alpha r^{\alpha-1})^2}} \ , \tag{7.8}$$

and the normalization (6.51)

$$\varphi^* \approx \frac{\sqrt{\mu} c_{\mu/\mu,\lambda} \sigma^*}{\sqrt{1 + (d\alpha r^{\alpha-1})^2}} \ . \tag{7.9}$$

The $R^{(\infty)}$ values from simulation results are used. For $N = 100$, they do not match so exactly. Equation (6.115) is not shown explicitly in the figure since its results were equivalent to (7.9) after normalization using (6.51).

The solid curve in the figure is obtained using (5.33) for the $D^{(\infty)}$-based approximation instead of $R^{(\infty)}$ from simulation results

$$\varphi^* \approx \frac{\sqrt{\mu}c_{\mu/\mu,\lambda}\sigma^*}{\sqrt{1+\left(\alpha d\left[\frac{(N-1)\sigma}{2\sqrt{\mu}c_{\mu/\mu,\lambda}}\right]^{\alpha-1}\right)^2}} = \frac{\sqrt{\mu}c_{\mu/\mu,\lambda}\sigma^*}{\sqrt{1+\left(\alpha\left[\frac{\sigma^*}{2\sqrt{\mu}c_{\mu/\mu,\lambda}}\right]^{\alpha-1}\right)^2}} \quad . \tag{7.10}$$

It serves as an evidence for the nonlinearity of $R^{(\infty)}$ and overestimates the $\hat{\varphi}^*$; however, $\hat{\sigma}^*$ is predicted correctly.

Additionally, the figure contains the dotted line labeled as "I". It is obtained using the surrogate mutation model for the conversion of φ values of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES to the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES (see Point 6.3.1.5). The mean values of the experimental results for the $(9/9_{\mathrm{I}}, 10)$-ES are transformed by dividing the σ values by $\sqrt{\mu} = 3$, and normalized in the same manner as done for the other curves. The data points obtained are connected by a dotted line. The results match to the ones obtained for the $(9/9_{\mathrm{D}}, 10)$-ES with diagonal $\mathbf{v}$, which can be considered as a verification of the applicability of the surrogate mutation model.

The values for σ^* and φ^* are quite low in the figure as compared to the ones in Figure 7.13. The purpose of this experiment was to show that the theoretical formulae obtained are also valid for such limit values of μ. The validity region of the simple local model is remarkable.

Table 7.1: The progress rate φ of the $(2/2_{\mathrm{D}}, 10)$-ES on the hyperplane with a diagonal $\mathbf{v}$ vector for various N, $\sigma = 1$. The (rounded) experimental mean for φ, its (rounded) standard error, the number of variables N, and the number of generations (samples) G are shown. For $N = 1$, one obtains $\varphi \approx c_{2,10}\sigma$; and for $N \to \infty$, $\varphi \approx \sqrt{2}c_{2/2,10}\sigma \approx 1.796\sigma$.

φ	s. e.	N	G
1.351	0.002	1	100 000
1.483	0.001	2	500 000
1.538	0.002	3	100 000
1.574	0.002	4	100 000
1.596	0.002	5	100 000
1.671	0.002	10	100 000
1.759	0.002	50	100 000
1.776	0.002	100	100 000
1.796	0.002	1000	100 000
1.795	0.002	10000	100 000

The $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES on the hyperplane. The progress rate of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES on the hyperplane was addressed in Point 6.3.1.6. The dependence of its progress rate values on the vector $\mathbf{v}$ can be investigated using simulation runs. The aim of these experiments is to show that the progress rate formulae in (6.59) hold for aligned and diagonal cases. Furthermore, the simulation runs are expected to show that these two cases of $\mathbf{v}$ yield two extremes for φ. In other words, for all other choices of $\mathbf{v}$, the progress rate should be between these two values.

Since the standard error of the empirical measurements were not small enough, it was not possible to show that these φ values are extreme: The lower limit ($\varphi = c_{\mu,\lambda}\sigma$) is clearly obtained for the aligned case. However, the experiments with randomly chosen $\mathbf{v}$ attained values rather close to the upper limit ($\varphi = \sqrt{\mu} c_{\mu/\mu,\lambda}\sigma$), some of them even contained this limit in their interval (mean $\pm$ standard error). These experiments were done for $N = 100$. Although there are 2^{100} different diagonal unit vectors $\mathbf{v}$ for this case, one can say that a lot of other vectors will give progress rate results in the vicinity of this upper limit.

Some simulations were made to investigate the progress rate values of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES on the hyperplane with a diagonal $\mathbf{v}$ vector. For $\mu = 2$, $\lambda = 10$, and $\sigma = 1$, the number of variables is increased from $N = 1$ up to $N = 10\,000$. Measured mean progress rates and respective standard errors can be found in Table 7.1.

The values of $c_{2,10}$ and $c_{2/2,10}$ can be found in Table 5.1 on Page 62. These results show that the surrogate mutation model holds for the hyperplane case if $\mathbf{v}$ is *diagonal* and if N is sufficiently large. Obviously, the value $\varphi = c_{\mu,\lambda}\sigma$ is obtained for the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES on the hyperplane if $\mathbf{v}$ is aligned, independent of the value of N.

7.2.8 The $(1, \lambda)$-ES for various α

The progress value of the $(1, \lambda)$-ES depends on the ridge function concerned. Therefore, it is important to describe the dependence of φ^* on α for a given set of σ^* values. At this point, it should be underlined that the σ values for different α may differ considerably from each other as a result of the normalization in (6.51), although the resulting σ^* value is identical. This emerges as a by-product of the generalization effect of the normalization. For $N = 100$ and $d = 0.01$, which were used in this experiment, the originating values for $\sigma^* = 1$ are $\sigma \approx 101$ for $\alpha = 1.5$ and $\sigma \approx 0.02$ for $\alpha = 8$.

In Figure 7.15, the normalized progress rates of the $(1, 10)$-ES for different α values are compared with each other. The values $\alpha \in \{1.5, 2, 3, 4, 8\}$ are used in the comparison. Additionally, the theoretical formula (6.45) using the $R^{(\infty)}$ values from simulations is drawn for these α values. Formula (6.89) for the parabolic ridge ($\alpha = 2$) gives similar results (not shown in the figure). Unfortunately, it cannot be generalized to other ridge functions since the $R^{(\infty)}$ formula required has not been derived for the general case. A good accordance of theoretical and empirical results can be observed. The deviation for the peak performance $\hat{\varphi}^*$ of $\alpha = 8$ vanishes if a larger N value is used in the experiments (e.g. $N = 1000$). The experiment for $\alpha = 8$ is repeated for $d = 2$. The resulting curve becomes equivalent to the one for $d = 0.01$ after the normalization in (6.51).

The theoretical progress rate on the hyperplane is shown as a line in the figure, $\varphi =$

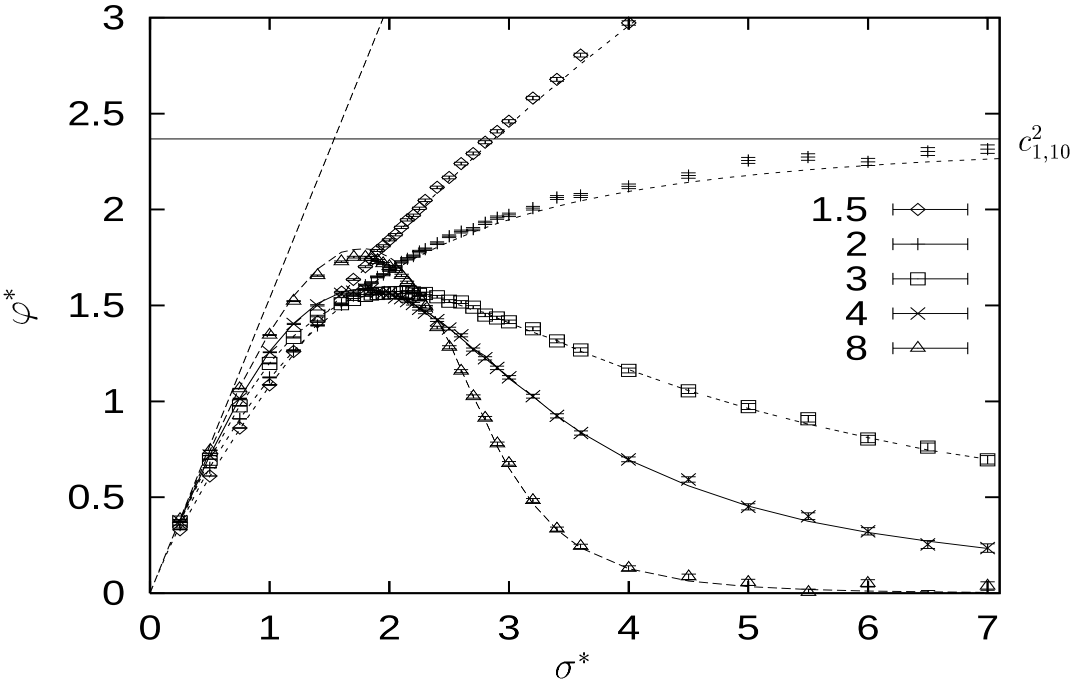

Figure 7.15: The progress rate φ versus the mutation strength σ of the $(1,10)$-ES on ridge functions, using (6.51) for the normalization. The experimental φ^* values are shown for $\alpha \in \{1.5, 2, 3, 4, 8\}$, along with corresponding analytical approximations obtained using (6.45) and $R^{(\infty)}$ values from simulations. The limit case $\alpha = 0$ with $\varphi^* = c_{1,10}\sigma^*$ and the asymptotic value $c^2_{1,10}$ for $\alpha = 2$ are also displayed.

$c_{1,10}\sigma$. One observes that this limit becomes active if α is increased. However, one has to consider that the actual σ values also decrease in this process. Additionally, one observes that the progress rate limit $\hat{\varphi}^* = c^2_{1,\lambda}$ for the parabolic ridge ($\alpha = 2$) seems to be an upper limit for other ridge functions with $\alpha > 2$. A proof for this observation is pending.

The difference in the progress behavior for $\alpha < 2$ and $\alpha > 2$ is clearly observed in this figure. The sharp ridge case ($\alpha = 1$) is not shown since it cannot be normalized using (6.51) and since the slope of its φ depends directly on the d value used in the fitness function. The case $\alpha = 1.5$ is plotted to indicate that its progress rate continues to increase if σ^* is increased, although not as "fast" as in the $\alpha = 0$ case, with the limit $\lim_{\sigma^* \to \infty} \varphi^* = \infty$. For $\alpha > 2$, one observes a sharper decrease in φ^* toward zero if α is increased; however, the progress rate values remain positive.

After considering the conjectures in Subsection 7.1.3, and the change in the φ^* curve for increasing α, one can infer how the φ^* curve will be for $\alpha \to \infty$. The $R^{(\infty)}$ value is expected to be slightly below 1 for $\sigma^* \lessapprox 2c_{1,\lambda}$. A linear increase and $R^{(\infty)} > 1$ is expected for larger σ^* values in φ^*, resulting a sharp decrease down to zero (see Equation (6.45)). For $\lambda = 10$, this sharp decrease is expected at $\sigma^* \gtrapprox 2c_{1,10} \approx 3$, and the φ^* values are expected to increase linearly for $0 < \sigma^* < 3$. Such a tendency is supported by the change of the φ^* curve for $\alpha \in \{3, 4, 8\}$.

If one would repeat the experiment in Figure 7.15 for the (μ, λ)-ES, the $(\mu/\mu_I, \lambda)$-ES, or the $(\mu/\mu_D, \lambda)$-ES, the tendencies of the curves will not be different. Similar curves can be obtained in various simulations if N is chosen sufficiently large.

7.2.9 The static progress rate on the ridge axis

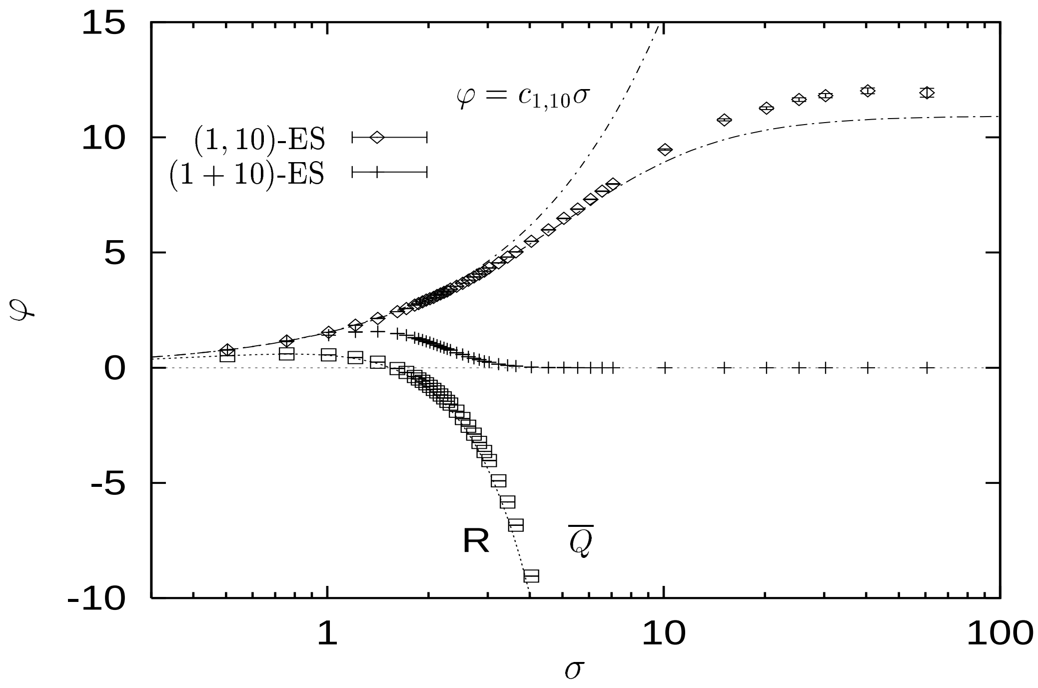

Figure 7.16: The static progress rate versus the mutation strength σ of the $(1 \overset{+}{,} 10)$-ES on the parabolic ridge for $r = 0$ (on the ridge axis). The formula (6.86) for the $(1, 10)$-ES accords to simulation results for small σ values, but the simulation results go to a different asymptotic limit for $\sigma \to \infty$. The experimental results of the $(1 + 10)$-ES are shown for comparison. The third set of simulation results represents the quality gain $\overline{Q}$ values of the $(1, 10)$-ES under the same conditions, and (5.37) perfectly matches to them ("R").

The static progress rate performance of the $(1 \overset{+}{,} \lambda)$-ES on the ridge axis will be investigated using the parabolic ridge. The simulation results of the $(1, 10)$-ES and $(1+10)$-ES for $r = 0$ will be used for this purpose. The static progress rate attains its maximum value for $r = 0$, and its limit for $\sigma \to \infty$ is expected to be much larger than the one for the stationary case (compare Equation (6.87) with Equation (6.90)). It is quite interesting that the progress rate φ is not expected to go to infinity for $r = 0$ and $\sigma \to \infty$. Furthermore, one expects remarkable differences between the values of quality gain $\overline{Q}$ and progress rate φ since the progress measure φ_R is definitely nonzero for $r = 0$ (see Equation (6.25)).

In Figure 7.16, three important series of experiments are depicted. From top to bottom, these are the experimental results of φ of the $(1,10)$-ES, φ on the $(1+10)$-ES, and $\overline{Q}$ of the $(1,10)$-ES. The progress rates for plus and comma cases strongly differ from each other. The approximation (6.86) accords to simulation results for lower values of σ; However, simulation results go to a larger asymptotic limit. Their accordance is expected to become better for larger values of N. This limit is much smaller than the limit $\varphi = c_{1,10}\sigma$ shown in the figure.

The progress rate results for the $(1+10)$-ES are much worse than the ones obtained for the stationary case (Figure 7.8). The reason is simple. The plus strategy can hardly create offspring which are better than their parent for $r = 0$. Almost all descendants are expected to have a worse fitness value than the parent. Therefore, $r = 0$ is not a good starting location for the elitist strategy. It is important to note that the experimental conditions were the same for the $(1+10)$-ES and $(1,10)$-ES, and these conditions harmed the former and favored the latter. This experimental setup is a good example for *biasing* the results in a certain way. The stationary comparison gives a more objective view for the evaluation of these strategies (see Subsection 7.2.1).

The third data series (lowest curve) gives experimental quality gain results of the $(1,10)$-ES. The formula (5.37) proposed by Rechenberg accords very well to them. It is important to note that this formula was proposed for the progress rate φ and not for the quality gain $\overline{Q}$. The quality gain formula (5.39) matches exactly to these simulation results, the values of its fitness-dependent parameters can be found in Point 6.1.2.2. This formula is omitted in the figure for clarity. After considering Table 6.1 and these fitness-dependent parameters, one notes that (5.37) emerges as a special case of (5.39) for small values of d, $r = 0$, and $N \gg 1$. The remaining difference (N instead of $N-1$) is negligible for $N \gg 1$. The terms with κ_3 and κ_4 vanish, one obtains $\overline{Q} \approx M_Q + c_{1,\lambda} S_Q$, However, these two formulae differ considerably for $d \gg 1$.

7.2.10 The static progress rate of some ridge functions

This subsection is dedicated to the static analysis of progress rate values of the $(1,\lambda)$-ES on different ridge functions. The ES algorithm with $\lambda = 10$ is investigated as an example on the sharp ridge ($\alpha = 1$), the parabolic ridge ($\alpha = 2$), and the ridge function with $\alpha = 8$, which has a highly nonlinear fitness landscape. The mutation strength σ is kept fixed, and the distance r to the ridge axis is varied for several orders of magnitudes.

Figure 7.17 summarizes the results for $\alpha \in \{1,2,8\}$. Both axes are logarithmic. For $\alpha = 8$, the mutation strength $\sigma = 0.034145$ ($\sigma^* \approx 1.75$) is chosen, since it yields the maximum stationary progress rate (see Figure 7.15). In the static case, it attains $\varphi \approx c_{1,10}\sigma$ for $r < 1$. For $r \gtrapprox 1$, however, a sharp decrease is observed on the progress rate values.

For the parabolic ridge case, the simulation is done for $\sigma = 2$. The maximum value $\hat{\varphi}_{st}$ is held for even larger values of r. The static progress rate decreases slower than in the $\alpha = 8$ case. For both $\alpha = 8$ and $\alpha = 2$, simulation results are correctly predicted by the static progress rate formula (6.84). However, this is not the case for the sharp ridge. It can be assumed that this general formula gives erroneous results for $\alpha < 2$ because of

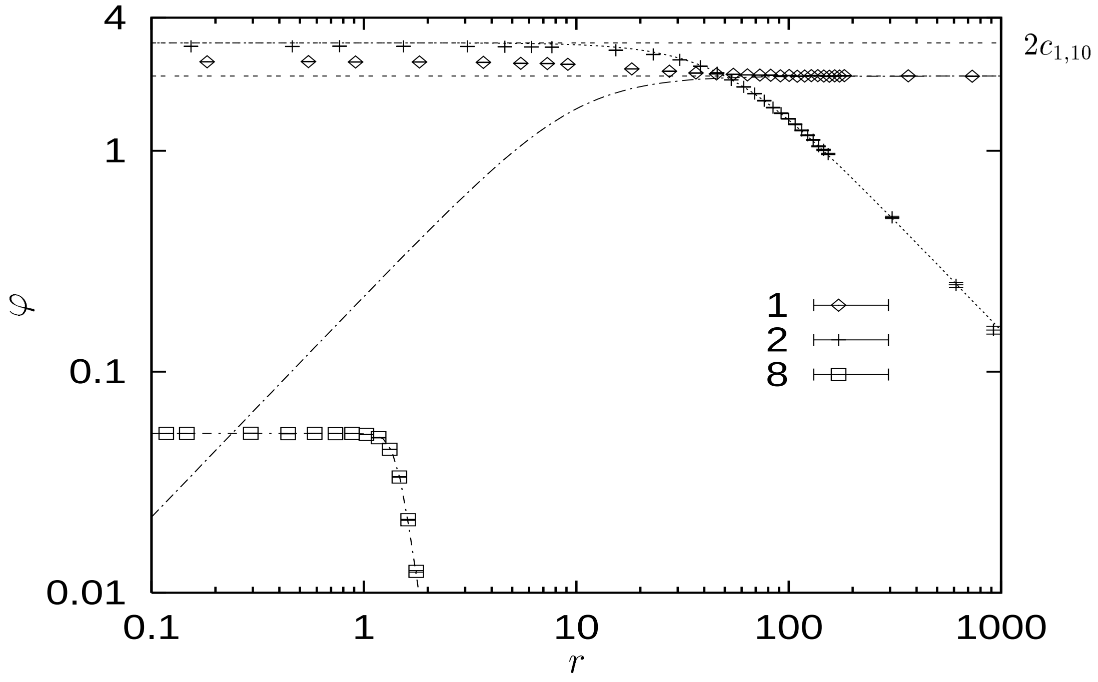

Figure 7.17: The static progress rate versus distance r to the ridge axis of the (1, 10)-ES on ridge functions with $\alpha \in \{1, 2, 8\}$. The value $\sigma = 0.034145$ is used for $\alpha = 8$. For other cases, $\sigma = 2$ is used. For the sharp ridge, $d = 1$ was chosen. The theoretical formula (6.45) is plotted for $\alpha = 2$ and $\alpha = 8$. Formula (6.84) gives wrong results for $\alpha = 1$. For this case, (6.52) gives a line.

the third term in the denominator. In this figure, the value $d = 1$ is arbitrarily chosen for the sharp ridge, and the lower limit of φ is correctly estimated by the general formula (6.84) and by (6.52). The theoretical results obtained from (6.45) are depicted for $\alpha = 2$ and $\alpha = 8$. This formula is much simpler than (6.84), and gives similar results in this figure. For $\alpha = 2$ and $r < 10$, Equation (6.84) predicts the simulation results better. It is interesting to see that the static progress rate of the sharp ridge depends relatively little on the r value, especially for larger r values it decreases a quite negligible amount. The local model promises no dependence on r, which is of course not the case.

Some more static results can be added here for the (1, 10)-ES on the parabolic ridge with $\alpha = 8$, $d = 0.01$. The plus strategy attains equivalent static results for $\sigma = 0.034145$. However, since its $R^{(\infty)}$ value is slightly smaller, the resulting stationary φ^* is larger for the elitist case. For $\sigma = 0.06829$, however, the result is more interesting. Although the comma strategy attains better static results for $1 \lessapprox r \lessapprox 3$ (the values are equivalent for other r), the stationary progress rate of the plus strategy is still better, because of the same reason (small $R^{(\infty)}$). The $R^{(\infty)}$ values of both algorithms are in the interval $1.5 \lessapprox R^{(\infty)} \lessapprox 2.5$.

The change in the static progress rate on the functions with $\alpha \geq 2$ for sufficiently high r values is remarkable. This change is even sharper if α is increased. As can be seen in this figure, the static progress rate formulae (6.45) and (6.84) successfully estimate the simulation results for $\alpha \geq 2$.

7.3 The quality gain $\overline{Q}$

The quality gain is analyzed only for the $(1, \lambda)$-ES and on the parabolic ridge function. The results can be generalized to the $(1 + \lambda)$-ES using the quality gain formula in [Bey96c, p. 119], [Bey94]; and to comma strategies other than $(1, \lambda)$-ES and to other ridge functions using the alternative approach in Subsection 6.1.4. The generalization of (5.39) to other ridge functions was discussed in Point 6.1.2.3. The quality gain is not analyzed further because of its significant difference from the progress rate φ on ridge functions: In general, the progress rate values cannot be obtained from the quality gain values. However, the alternative approach to the quality gain has shown that this measure in the fitness space can be calculated using two appropriate progress measures (φ and φ_R) in the search space.

This subsection has two parts. In the first one, the dependence of the quality gain on r values and its principal difference to the progress rate φ is shown. Additionally, the approximation quality of the formulae (5.39), (6.25), and (7.11) are shown for the $(1, \lambda)$-ES on the parabolic ridge in Figure 7.18. Secondly, the values of $\overline{Q}$ and φ are compared for the $(1 \overset{+}{,} \lambda)$-ES under the same conditions, showing that such a characteristic difference between these measures also exists for the plus strategy. In Subsection 7.3.2, it will also be shown that plus and comma strategies can behave similarly or differently according to these two progress measures, depending on the distance r to the ridge axis.

The measurements are made statically for a constant mutation strength σ and varying distance r to the ridge axis. This is a relatively new analysis method, since the reverse way (keeping r constant and varying σ) was popular in the theoretical analysis of the ES on the sphere model.

As can be seen in Subsection 7.3.2 or in the alternative approach to the $\overline{Q}$ analysis, the quality gain is expected to give similar results only in the stationary case. Two conditions for this case ($\overline{Q} \approx \varphi$) were mentioned in Section 6.1 (Page 77 and Page 79), and further additional conditions may exist. For example, the fitness function may represent a real world problem. In this case, the fitness function values and object variables have different units. As a result, the quality gain and progress rate have different units, too. Therefore, the even if the magnitudes of $\overline{Q}$ and φ can be equivalent, their units will never be.

Although the mean values of $\overline{Q}$ and φ are similar in the stationary case, the expected values of their second moments differ. As a result, in stationary measurements, the $\overline{Q}$ values possess much larger standard errors than φ. This difference increases further if the mutation strength is increased, and the standard error of the former one increases with an higher order than the latter. The stationary analysis of $\overline{Q}$ is therefore omitted.

7.3.1 The static quality gain $\overline{Q}$

This subsection aims at the verification of $\overline{Q}$ formulae and the investigation of the static $\overline{Q}$ behavior. The $(1, 10)$-ES algorithm is analyzed on the parabolic ridge for this purpose.

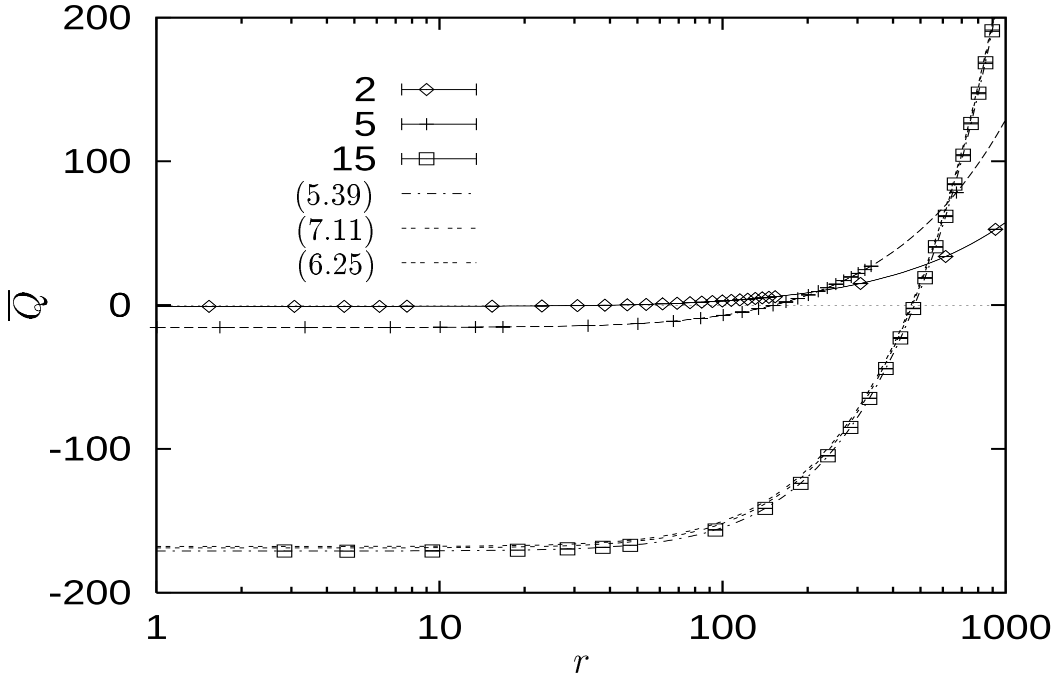

Figure 7.18: The quality gain $\overline{Q}$ of the (1, 10)-ES on the parabolic ridge for $\sigma \in \{2, 5, 15\}$ at different values of r. These static experimental results are compared with (5.39). For $\sigma = 15$, this formula is compared with two further formulae, (7.11) and (6.25).

The values $\sigma \in \{2, 5, 15\}$ are used for the mutation strength.

Figure 7.18 indicates that quality gain values are negative for small r values, and that they are at a smaller level for larger σ. These values would be positive for smaller σ values (not shown here). Remarkably, progress rate values of ES algorithms become never negative on ridge functions (see Section 7.2 and also the formulae in Section 6.3). Moreover, progress rate values attain their maximum value for small r (see Figure 7.17).

For larger values of r, the $\overline{Q}$ values increase, become positive, and go to infinity. A sketch of the proof for this observation ($\lim_{r\to\infty} \overline{Q} = \infty$) on the parabolic ridge can be stated easily. Equation (6.25) and the theoretical and empirical results obtained from the static analysis of φ and φ_R will be used for this purpose. As r goes to infinity, φ goes to zero (see Figure 7.17), and φ_R goes to $c_{1,\lambda}\sigma$ (see Figure 7.5). Consequently, the second term in (6.25) goes to infinity for $r \to \infty$, whereas the first one goes to zero and the third one asymptotically goes to a constant value. In order to generalize this result for $\alpha > 2$, the asymptotic ($r \to \infty$) values of φ and φ_R are used. The alternative $\overline{Q}$ formula (6.25) has a different form for $\alpha \neq 2$. However, $\overline{Q}$ can still be expressed using φ and φ_R. The former one is zero (e.g. see Figure 7.17 for $\alpha = 8$, or see Equation (6.84)). The latter one (φ_R) goes to the upper progress rate limit ($c_{1,\lambda}\sigma$) because of the local curvature conditions. Consequently, one obtains the same limit ($\lim_{r\to\infty} \overline{Q} = \infty$). As a conjecture, this result can be extended to other ridge functions with $1 < \alpha < 2$. As a side remark, $\overline{Q}$ and φ may also differ for the sphere model.

Another observation is also important: The slope of the $\overline{Q}$ curve increases with σ. In contrast to the limit value of $\overline{Q}$ for $r \to \infty$, the progress rate φ of ES algorithms on the parabolic ridge does never go to infinity for constant σ. For the conditions in Figure 7.18, static φ values of ES algorithms with comma strategy decrease (and not increase) as r is increased. As a result, one can conclude that the progress measures $\overline{Q}$ and φ behave inversely for this static experiment.

The second aim of this experiment is the verification of quality gain formulae. Equation (5.39) perfectly describes the simulation results. For $\sigma = 15$, two other formulae are tested additionally. The formula

$$\overline{Q} = M_Q - \frac{\kappa_3}{6} S_Q + \left[1 + \frac{5}{36}\kappa_3^2 - \frac{\kappa_4}{8}\right] c_{1,\lambda} S_Q \tag{7.11}$$

is obtained from (5.39) by neglecting $d_{1,\lambda}^{(2)}$ and $d_{1,\lambda}^{(3)}$ terms. It yields slight and tolerable deviations from simulation results for small r. The error made by this approximation naturally increases with σ. For sufficiently large r, the difference between (5.39) and (7.11) is negligible. Similarly, the alternative $\overline{Q}$ formula (6.25) is also depicted for this case. This formula gives quite accurate results, too.

One observes a sharp increase in $\overline{Q}$ values in the neighborhood of $R^{(\infty)}$ (see Subsection 7.1 for $R^{(\infty)}$ values). For $r \approx R^{(\infty)}$, one expects $\overline{Q} \approx \varphi$. If the analytical $R^{(\infty)}$ formula is used, the deviation from the theoretical formula from the actual $R^{(\infty)}$ values can cause wrong predictions of the stationary $\overline{Q}$. Since the $R^{(\infty)}$ formula (6.166) underestimates the actual stationary value, the $\overline{Q}$ formula may even give negative results for the stationary case. The $\overline{Q}$ formula gives the correct stationary value only if accurate $R^{(\infty)}$ values (e.g. obtained from simulation results) are provided. The stationary φ formulae are immune to such errors in the $R^{(\infty)}$ formula.

7.3.2 Progress measures in comparison

The static progress measures of the $(1,10)$-ES and $(1+10)$-ES are compared in this subsection. The mutation strength $\sigma = 2$ is used for both algorithms. At this σ value, the stationary values of φ and $\overline{Q}$ on the $(1,10)$-ES and $(1+10)$-ES only differ slightly from each other (see Figure 7.8). Parameters other than σ are also the same for these algorithms, i.e. the comparison is made under the same conditions.

This figure is important because it shows the differences in the static behavior of $(1 \overset{+}{,} \lambda)$-ES algorithms. Furthermore, it underlines the differences in the r-dependence of progress measures in the search space and fitness space. Additionally, it contains the stationary case, and explains how both progress measures can give the same value (see Subsection 7.1.5 for an alternative explanation). It stresses an important difference in the way φ and $\overline{Q}$ values change with r for the $(1,\lambda)$-ES on the parabolic ridge (and also for $\alpha > 2$). As r increases, φ decreases and $\overline{Q}$ decreases.

First of all, both algorithms attain similar φ values for sufficiently large r and the progress rate curves go to zero. The $\overline{Q}$ values attain a quite large slope, and go to infinity.

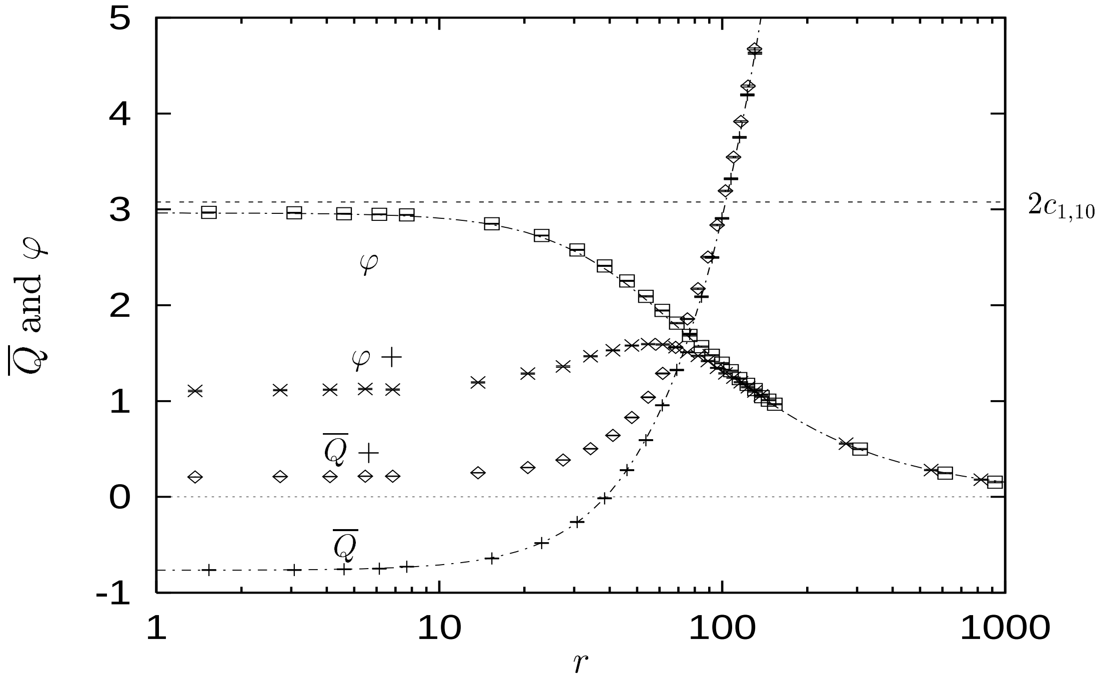

Figure 7.19: The quality gain $\overline{Q}$ and the progress rate φ of the $(1 \overset{+}{,} 10)$-ES on the parabolic ridge. The static results are depicted for $\sigma = 2$ and various r values. The elitist strategy (labeled with an additional "+") attains larger $\overline{Q}$ values and smaller φ values. The φ formula (6.86) and $\overline{Q}$ formula (5.39) are plotted, as well as the limit $\varphi = 2c_{1,10} \approx 3.1$.

The result for the stationary case is included in this figure (at $r \approx R^{(\infty)}$). For this r value, the curves for $\overline{Q}$ and φ of both algorithms intersect each other. The $R^{(\infty)}$ value and the stationary value $\varphi \approx \overline{Q}$ are larger for the $(1, 10)$-ES compared to the $(1 + 10)$-ES.

For smaller r values, these four curves differ from each other. One can clearly see that the φ curve of the $(1, 10)$-ES increases to an asymptotic value and matches the theoretical formula (6.86). Its limit value for $r \to 0$ is smaller than the horizontal line $\varphi = c_{1,10}\sigma$, i.e. the progress rate of the hyperplane. For larger values of σ, the simulation results for this limit are underestimated by this formula. However, they also do not become as large as the maximum progress rate $\varphi = c_{1,10}\sigma$ of the $(1, 10)$-ES (see also Figure 7.16).

The plus strategy attains smaller progress rates, and its static φ values even decrease to an asymptotic value as $r \to 0$. As can be seen in Figure 7.16, this limit value can even be zero.

If the comparison is done using quality gain values (instead of φ values), the performance order of these ES algorithms is reversed. The quality gain of plus strategy is larger, and always positive. It decreases down to zero as $r \to 0$ for larger σ values. The quality gain $\overline{Q}$ of the $(1, 10)$-ES becomes negative for $r \to 0$. Detailed explanations on $\overline{Q}$ of the $(1, \lambda)$-ES can be found in Subsection 7.3.1. The mean values (of the simulation results for $\overline{Q}$ and φ) in this subsection were used in [OBS98c] to obtain a similar figure; however, no theoretical results were supplied there.

7.4 The success probability P_{s1}

The formulae for the general ridge function (6.40) and for the parabolic ridge (6.33) will be tested under three different experimental setups. The relation between the success probability and the progress rate will also be investigated and discussed. The first two experiments are on the parabolic ridge. The first one is done statically at $r = 0$ for various σ values, whereas the second one is stationary. The last experiment aims at the P_{s1} curves on three different fitness functions: The sharp ridge, parabolic ridge, and the case for $\alpha = 8$. Additionally, the P_{s1} values of the plus strategy are compared to the ones of the $(1, \lambda)$-ES on the parabolic ridge under the same conditions in Subsection 7.4.2.

7.4.1 On the ridge axis of the parabolic ridge

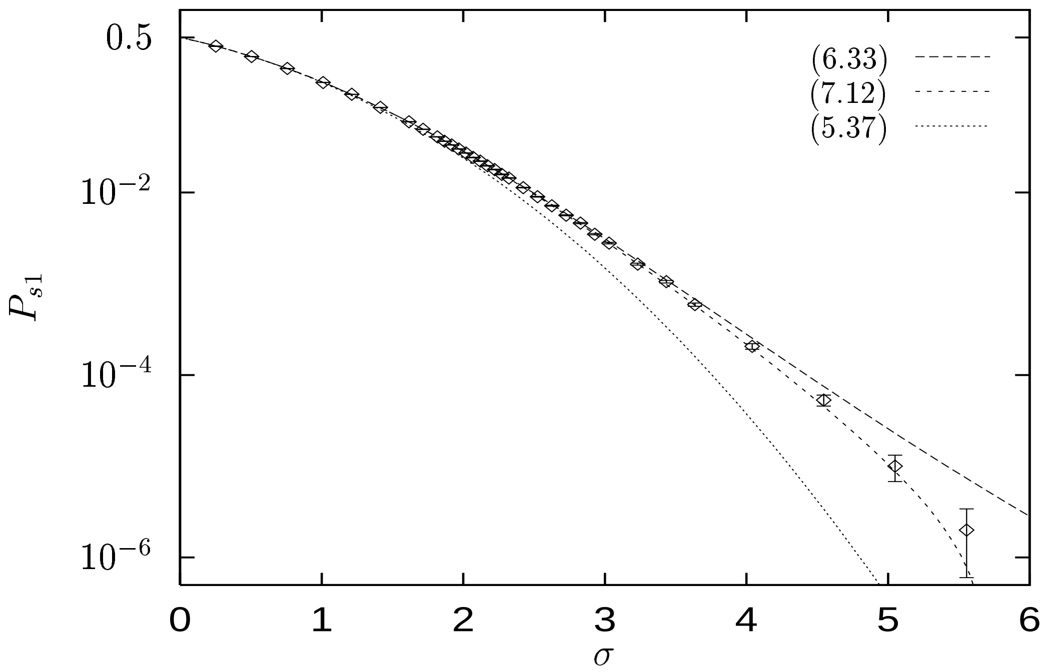

Figure 7.20: The success probability P_{s1} versus the mutation strength σ of the $(1, 10)$-ES on the parabolic ridge. The static results for $r = 0$ (i.e. on the ridge axis) are depicted (noticed by "$\diamond$"). The simulation results are estimated by (6.33) and then by (7.12). Rechenberg's formula (5.37) [Rec94, p. 66] gives also useful results.

This subsection aims at the comparison of static success probability formulae with simulation results. The $(1, 10)$-ES is statically investigated on the parabolic ridge at $r = 0$

(on the ridge axis). The progress rate values for this experiment can be found in Figure 7.16.

In Subsection 6.2.1, the first order approximation to the success probability was given by (6.33). It was mentioned that this P_{s1} approximation can be made more accurate if necessary. The extreme case $r = 0$ necessitates such a correction, since (6.33) is not accurate enough. As it was described in that subsection, the second order approximation is obtained using (6.32), (6.30), and (6.31) as

$$P_{s1} \approx \Phi(z) + \frac{1}{\sqrt{2\pi}} \frac{\kappa_3}{3!} \mathrm{e}^{-\frac{1}{2}z^2}(z^2 - 1) \ , \tag{7.12}$$

where z stands for (see (6.32) and (6.33))

$$z = \left. \frac{Q - M_Q}{S_Q} \right|_{Q=0} = -\frac{(N-1)d\sigma}{\sqrt{1 + (2dr)^2 + 2d^2(N-1)\sigma^2}} \ . \tag{7.13}$$

The values of M_Q, S_Q, and κ_3 can be found in Point 6.1.2.2. One has to substitute $r = 0$ for this case. As can be seen in Figure 7.20, the second order approximation is more accurate than the first one. The simple formula (5.37) proposed by Rechenberg underestimates the simulation results although it gives reasonable results for small values of the mutation strength σ. If one compares the complexity of the second order approximation (7.12) with the one of (5.37), one may prefer the latter because of its simplicity and shortness. As a side remark, (5.37) emerges as a special case of (6.33) if the denominator is approximated by 1 for $r = 0$ and $N \approx N - 1$.

The success probability P_{s1} becomes zero for $\sigma \to \infty$ at $r = 0$. One should note that the maximum static progress rate is obtained on the parabolic ridge at the minimum value (zero) of the success probability P_{s1}. This will also be confirmed for this fitness function in the following two subsections by experiments.

7.4.2 Stationary values on the parabolic ridge

The success probability values of the $(1, \lambda)$-ES and the $(1 + \lambda)$-ES are compared in Figure 7.21 for $\lambda = 10$ descendants. The results are obtained for the stationary case on the parabolic ridge. The simulation results for $R^{(\infty)}$ and φ^* obtained using the same experimental setup can be found in Subsection 7.1.1 and Subsection 7.2.1, respectively.

The P_{s1} values of the $(1 + 10)$-ES become definitely smaller than the non-elitist case for larger values of σ^*. They decrease down to zero although the P_{s1} values of the $(1, 10)$-ES converge to an asymptotic value. The values of the non-elitist case are correctly approximated by (6.35). Its asymptotic ($\sigma^* \to \infty$) limit is given by (6.36). This limit is $\Phi(-c_{1,10}) \approx 0.062$ for $\lambda = 10$. It accords to the simulation results. Interestingly, the maximum stationary progress rate value of the $(1, 10)$-ES is attained where the success probability P_{s1} is at its minimum. This P_{s1} limit decreases further for increasing λ, although the corresponding progress rate limit increases (see Equation (6.90)).

It was conjectured that the measured success probability values can be used for obtaining the optimum mutation strength (see Section 2.6.11 and Page 52). The 1/5-th success

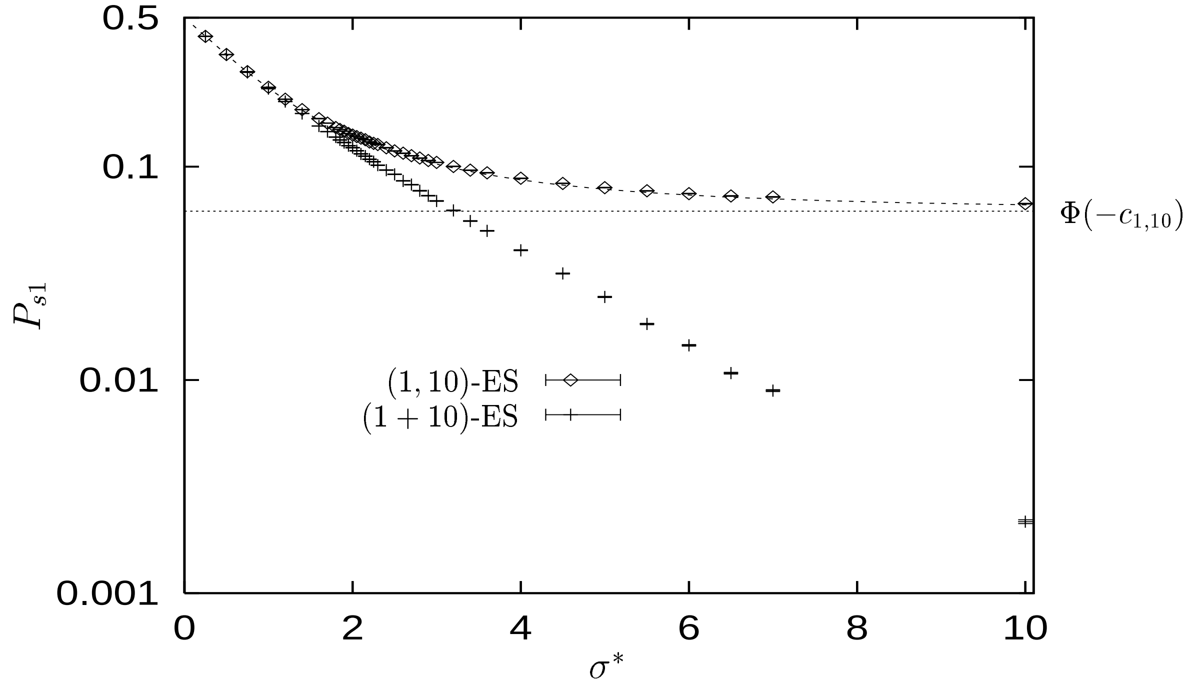

Figure 7.21: The success probability versus σ^* for the $(1 \overset{+}{,} 10)$-ES on the parabolic ridge. Equation (6.48) is used for the normalization of the mutation strength. The stationary P_{s1} (6.35) of the $(1, \lambda)$-ES and its asymptotic value $\Phi(-c_{1,10})$ for $\lambda = 10$ are shown.

rule [Rec71] is a result of the analysis on the corridor model and the sphere model for such a relation. After investigating Figure 7.21, one notes that the P_{s1} value at the optimum mutation strength is much different than the one on the sphere model and on the corridor model. Figure 7.17 and Figure 7.22 can be used to generalize this result. Therefore, it can be said that no general rule can be devised using the P_{s1} values for predicting σ^* at which the maximum progress rate value is attained. Furthermore, the other success measure $P_{s\lambda}$ does not help more than P_{s1} in finding the optimum mutation strength, since it is just a linearly increasing function of P_{s1} (see (4.10)). That is, both P_{s1} and P_{s1} cannot help for running the ES algorithm at the optimum progress rate.

7.4.3 Static values of three ridge functions

This subsection investigates the success probability P_{s1} for the experimental setup in Subsection 7.2.10. The parameters used for each fitness function, the mutation strengths used, as well as further explanations can be found there. Furthermore, this subsection aims at the investigation of the relation between P_{s1} and φ.

Each of these three P_{s1} curves in Figure 7.22 has a different characteristic. An increase in d shifts the curve for the sharp ridge to the right, a decrease to the left. The same change is observed after varying σ. For the parabolic ridge case, an increase in d or σ causes a

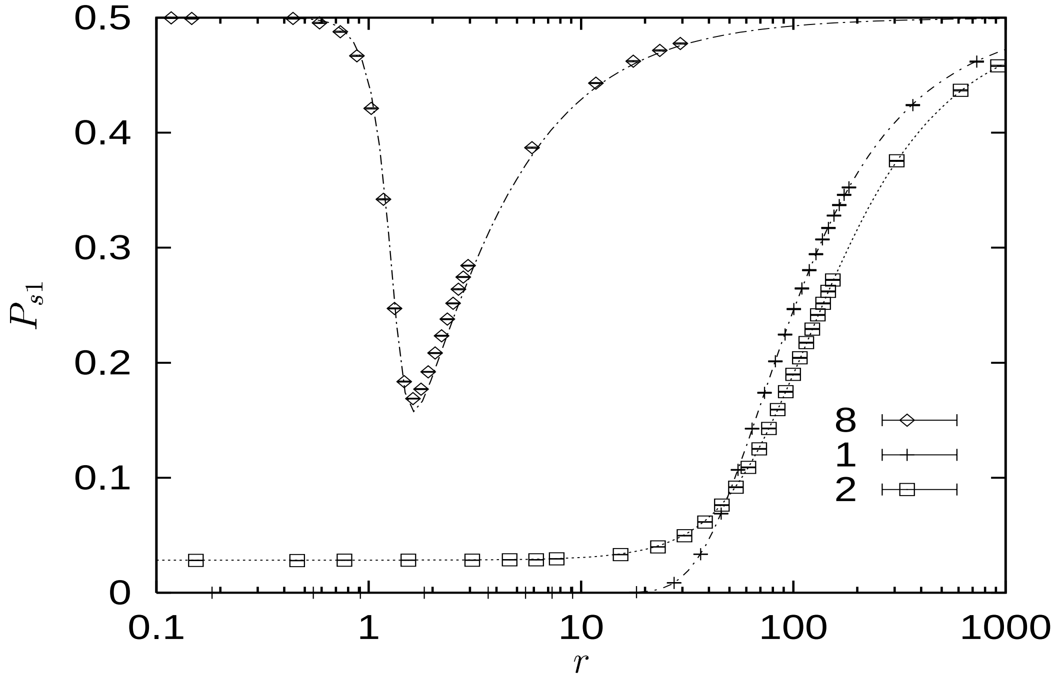

Figure 7.22: The success probability P_{s1} versus r for the (1, 10)-ES on the sharp ridge ($\alpha = 1$, $d = 1$), parabolic ridge ($\alpha = 2$), and for the $\alpha = 8$ case. The mutation strength was $\sigma = 2$ for the first two cases, and $\sigma \approx 0.034$ for the last case. The theoretical formula (6.40) and its special case (6.33) for the parabolic ridge are displayed.

decrease in the asymptote obtained for $r \to 0$ down to zero; however, the curve does not significantly shift to the right or left. An increase in σ for the $\alpha = 8$ case causes the curve to decrease down to zero, and to hold this value for a longer interval of r. However, the main characteristics of these three figures do not change.

The increase of P_{s1} values for $\alpha = 8$ as $r \to 0$ can simply be explained by the shape of isofitness lines (see e.g. Figure 3.7 for the $\alpha = 10$ case). For $r \to 0$, the local curvature of the isofitness lines is so small that this fitness function yields P_{s1} values comparable to the ones of the hyperplane.

For the parabolic ridge and sharp ridge, the maximum φ value is obtained where P_{s1} is at its minimum value. Therefore, one may conclude that maximum static progress rate is obtained at minimum P_{s1} value. However, this conclusion has no predictive power, since the minimum of P_{s1} for $\alpha = 8$ guides to an r value which is irrelevant for maximizing the static progress rate. Another aim of this figure is the verification of theoretical formulae: It is nice to see that the simple formula (6.40) for the success probability and its special case (6.33) for the parabolic ridge ($\alpha = 2$) gives useful results.

7.5 Conclusions

The theoretical and experimental results have been compared in this chapter. Using simulations, it has been possible to show the correctness of analytical approximations derived in the previous chapter, as well as to compare (empirical) results for different ES algorithms or on different fitness functions. Static and stationary analysis have been used to investigate different aspects of convergence measures. Moreover, dynamic analysis has been used for the distance r to the ridge axis.

The main goal of this chapter has been the experimental verification of the formulae derived in Chapter 6. Additionally, the progress rate of the elitist strategy has been compared with the non-elitist one using $(1 \overset{+}{,} 10)$-ES on the parabolic ridge. For the stationary case, both $\overline{Q}$ and φ values of the $(1+10)$-ES are smaller. For the static case, its progress rate is smaller, and its quality gain larger. These experiments have been used to interpret the hypothesis on the elitist EA in Subsection 5.2.3. This hypothesis states the superiority of the improvement attained by the elitist $(1+\lambda)$-EA over the non-elitist $(1,\lambda)$-EA.

Some interesting observations will be outlined below.

1. A local model was proposed in Subsection 6.3.1 for the investigation of the progress rate φ in the stationary case or for large r. It gives also satisfactory results for the static case (for $\alpha \geq 2$).

2. The progress rate of the $(1,\lambda)$-ES on the ridge axis (i.e. $r=0$) of the parabolic ridge goes to a finite nonzero limit as the mutation strength σ goes to infinity ($\lim_{\sigma\to\infty} \neq \infty$ for $r=0$ and $\alpha=2$). This result can also be extended to other non-elitist ES algorithms for this fitness function.

3. The static comparison of the $(1 \overset{+}{,} \lambda)$-ES on the ridge axis is an interesting example of biasing the simulation results. This condition ($r=0$) favors the $(1,\lambda)$-ES and harms the $(1+\lambda)$-ES.

4. The surrogate mutation model for the $(\mu/\mu_{\mathrm{D}},\lambda)$-ES holds for the ridge functions if the progress direction $\mathbf{v}$ is chosen to be diagonal. The largest deviations from the formulae obtained using this model have been observed for the aligned $\mathbf{v}$ vector.

5. The genetic repair hypothesis holds for ridge functions ($\alpha > 1$) in an interesting form: The non-elitist algorithms with recombination satisfy the short term goal better and attain smaller $R^{(\infty)}$ values. As a result, they fulfill the long term goal better, and attain larger progress rate values.

6. Elitist algorithms fulfill the short term goal better than their non-elitist counterparts. However, this achievement alone does not help for a larger progress rate φ.

7. The progress rate φ and the quality gain $\overline{Q}$ behave inversely for the static analysis. This result has been obtained on the parabolic ridge ($\alpha=2$). However, it should be generalizable to $\alpha > 1$. Therefore, the quality gain cannot be used in general in estimating the progress rate.

8. For the parabolic ridge and the sharp ridge (i.e. $\alpha = 2$ and $\alpha = 1$), the maximal progress rate for the $(1, \lambda)$-ES is attained where the success probability P_{s1} has its minimal value. This result has been obtained in both static and stationary cases. Therefore, the success probability P_{s1} (or similarly $P_{s\lambda}$) has not been helpful in estimating the optimum mutation strength for the maximal progress rate.

9. An interesting relation (7.5) on progress coefficients of the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES has been obtained. Another relation (7.4) emerges as a special case of it. Such relations help in the preparations of the progress coefficient tables.

Two further results will be added to these conclusions. They are not directly related to any of the sections of this chapter, but they should be mentioned for the sake of completeness. The first one is on the MISR hypothesis (see Subsection 5.2.8). This hypothesis asserts that the individuals generated by the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES do not diffuse arbitrarily in the search space even if the selection operator is switched off. The standard deviation of the parents around the centroid is expected to be about $\sqrt{\mu}\sigma$, if it is averaged over generations. This value is observed in the experiments on the hyperplane for the selection-invariant variables in the search space. This is the case even after the rotation of the direction vector $\mathbf{v}$ for sufficiently large N ($N \gtrapprox 1000$).

Secondly, the hypothesis that corridor models (see Figure 3.8 in Subsection 3.3.3) may serve as the limit case ($\alpha \to \infty$) for ridge functions [Sch97] was tested. In [OBS98a, pp.35-41], it was shown using stationary experiments that this is not the case for the rectangular corridor and cylindrical corridor. The static analysis is carried out thereafter, with the result that the mechanisms yielding the $R^{(\infty)}$ value are entirely different for ridge functions and corridor models. As a result, one has different tendencies for φ_R, and for $R^{(\infty)}$ at large σ in the stationary case. Another important point is that the *algorithm* must be different for these cases. Two different algorithms generally cannot be compared on two different functions, i.e. a comparison after changing two basic factors is not reliable. A small change must be accomplished on the ES algorithm before running it on corridor models. The descendants generated outside the corridor are rejected, and one has to continue to generate indefinite number of offspring (denoted by $\lambda^{'}$, $\lambda^{'} \geq \lambda$) *until* one has λ feasible individuals (see Subsection 2.6.3). This remedy for invalid descendants reflects itself in the progress rate calculation: One calculates the progress rate for these λ offspring (which is basically the progress rate for the hyperplane), and multiplies the quantity obtained by the factor $\lambda/\lambda^{'}$. This factor is responsible for the performance decrease on corridor models. However, the decrease in φ on ridge functions with large α for $\sigma \to \infty$ is explained by an entirely different mechanism (the simple local model). Because of these principal reasons, simulation results obtained for the cylindrical corridor are omitted in this work.

Chapter 8

Summary and Discussion

The convergence behavior of evolution strategies (ES) on ridge functions has been investigated in this work. The term "convergence behavior" covers the static, stationary and dynamic analysis of ES algorithms in the space of object variables (the search space). The analysis is primarily carried out using the measures defined in the search space (i.e. φ and r). The results have been derived for the $(1, \lambda)$-ES, the (μ, λ)-ES, the $(\mu/\mu_{\mathrm{I}}, \lambda)$-ES, and the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES. The progress rate φ measures the expected progress in the search space toward the optimum in one generation. The symbol r stands for the distance to the ridge axis. Its stationary value $R^{(\infty)}$ and other formulae necessary for its static and dynamic analysis have been derived. In this work, maximizing the progress rate φ is also termed as the long term goal, and the minimization of r as the short term goal, respectively.

The analytical derivations of the formulae of the search space measures can be found in Chapter 6. These formulae are asymptotically ($N \to \infty$) exact. Using simulations, it has been shown that these formulae yield satisfactory results for the finite number of variables N.

Additionally, the formulae for the fitness space measures $\overline{Q}$, P_{s1}, and $P_{s\lambda}$ have been derived for the $(1, \lambda)$-ES. The quality gain $\overline{Q}$ measures the expected progress in the fitness space. It has been proposed to estimate the progress rate φ. The success measure P_{s1} gives the probability of generating a descendant with a fitness value better than the average fitness value of the previous generation. The success probability $P_{s\lambda}$ is defined analogously for all λ descendants. It has been also investigated if P_{s1} and $P_{s\lambda}$ can be used to estimate the optimal mutation strength.

8.1 Conclusions

This section summarizes the striking and important results achieved in this work. Some further conclusions can be found in Point 6.3.1.8, in Subsection 6.3.7, in Section 6.5, and Section 7.5. A few research directions and some unanswered questions are outlined in Section 8.2.

1. The experimental results have shown that for ridge functions $\overline{Q}$ and φ possess entirely

different characteristics. Therefore, the quality gain $\overline{Q}$ cannot be used in general to estimate φ.

2. In general, the success measures P_{s1} and $P_{s\lambda}$ cannot be used to estimate the optimal mutation strength. Therefore, they are also not much helpful in maximizing the progress rate φ, either.

3. The fitness space measures ($\overline{Q}$, P_{s1}, and $P_{s\lambda}$) cannot be used (on ridge functions) to estimate the search space measures (φ and the measures related to r).

4. It has been shown that the ES algorithms do not diffuse along the gradient path on the ridge functions (see Page 55). The component of the mutation vector orthogonal to the direction given by the gradient is much larger than the one in the gradient direction.

5. In the "evolution window" hypothesis of Rechenberg (see Page 55), it was conjectured that a positive progress rate φ is only attainable for a finite interval of the mutation strength σ. The results of this study show that this interval is *not* finite for the ridge functions with $\alpha \leq 2$.

6. The universal progress law (see Page 56) states that the progress rate φ should be negative for $\sigma \to \infty$. This is not the case for the ridge functions for any α.

7. The maximum progress rate is obtained on the hyperplane ($\alpha = 0$) for a given ES algorithm. Therefore, it can be considered as a limiting case of fitness landscapes.

8. The progress rate φ of elitist strategies is not necessarily better than the non-elitist ones. Furthermore, the stationary quality gain $\overline{Q}$ of the elitist strategy (e.g. the $(1 + \lambda)$-ES) can be worse than the non-elitist counterpart (i.e. the $(1, \lambda)$-ES) even on multimodal functions.

9. Measuring the progress in fitness values can be misleading (even in unimodal functions). The local measurements indicating large fitness increase do not necessitate large progress in the search space toward the optimum. Moreover, the elitist strategies may achieve smaller progress values since they reject offspring with fitness values worse than the parents'.

10. The evolutionary progress principle (EPP) and the genetic repair hypothesis (GR) were developed as a result of the research on the sphere model. They also hold for ridge functions. They are used to explain an interesting stationary observation: Since recombination yields a smaller stationary distance $R^{(\infty)}$ to the ridge axis, the resulting progress rate is larger.

11. Depending on the value of α, one obtains various characteristics for the convergence behavior (especially for the progress rate φ) of ES algorithms. Because of this property, the ridge functions are expected to model the convergence behavior of ES algorithms on a variety of fitness landscapes.

12. Since the optimum of ridge functions is at infinity (see (3.18)), the number of generations necessary to reach the optimum cannot be determined. Therefore, an order estimation is not possible for the optimization algorithms.

8.2 Future research

Several important tasks need to be examined in the near future. Many of them have been mentioned in respective chapters. One can find the pages in which they appear under the index item "open problems". A few important ones are summarized here. Additionally, some interesting research directions are also presented.

1. The formulae derived in this work are asymptotically ($N \to \infty$) exact. For finite number of variables N, they differ from the simulation results, although they mostly give satisfactory results. The error in these predictions is not formally determined yet. Additionally, the N-dependent versions for these formulae are needed to be derived.

2. The formulae related to the distance r to the ridge axis (e.g. the stationary $R^{(\infty)}$ value, the time constant ω, φ_R, etc.) have only been derived for the parabolic ridge. The general case for any ridge function has not been derived yet.

3. The maximal normalized progress rate value was denoted as $\hat{\varphi}^*$. For $\alpha > 2$, this value as well as the mutation strength yielding $\hat{\varphi}^*$ are unknown. The general $R^{(\infty)}$ formula should be derived first for this investigation.

4. The formulae for the progress rate and the stationary $R^{(\infty)}$ for the (μ, λ)-ES are obtained by a conjecture. Although they predict the simulation results correctly, a proof may be necessary to show why and under which conditions the formulae for the (μ, λ)-ES can be obtained from the respective formulae of the $(1, \lambda)$-ES.

5. The analysis of the $(\mu/\rho, \lambda)$-ES is still pending, which should show whether $\rho < \mu$ can yield larger progress rates under some conditions. This analysis is assumed to be at least as hard as the analysis of the (μ, λ)-ES.

6. As a conjecture, one can propose the ridge functions for modeling the fitness landscapes distant from the optimum. This hypothesis seems to be plausible as long as the progress rate is positive. It must be reformulated more precisely: Especially on multimodal functions, it is not expected to be valid.

7. The application of the local model and induced order statistics on other fitness functions needs further attentions. The convergence behavior of ES algorithms on other fitness landscapes can also be investigated using these models.

8. The effectiveness of the success probability (P_{s1} and $P_{s\lambda}$) in estimating the optimum value of the mutation strength σ for the maximum progress rate φ is still unanswered. The analysis on the ridge functions showed that the success probability values are not conclusive. Therefore, they should be used with care in searching for the optimal mutation strength value. The investigation is expected to continue on other fitness functions.

9. The experimental values of the quality gain $\overline{Q}$ and the progress rate φ can be obtained on any function. These values can be used in the experimental analysis, and in the analysis of the convergence behavior, e.g. on multimodal landscapes.

10. The comparison of the $(1, \lambda)$-ES and the $(1 + \lambda)$-ES can also be carried out on multimodal landscapes. Especially the local measurements of the progress measures $\overline{Q}$ and φ at a local optimum may give interesting results on the convergence behavior of these algorithms.

11. The surrogate mutation model for explaining the functioning mechanism of the $(\mu/\mu_{\mathrm{D}}, \lambda)$-ES has an hypothetical character. First, the necessary conditions for its validity should be investigated. Secondly, alternative explanations are desired at least for the cases where this model does not hold.

12. The progress efficiency η (see Page 113) was introduced to measure the progress attained per descendant generated and to compare the progress rate values of different algorithms. For $\alpha > 2$, the optimum value of λ is unknown for the $(1, \lambda)$-ES.

13. The progress coefficients (see Page 61) are obtained using numerical integrations. Any method for simplifying these calculations or the relations between the progress coefficients are helpful in obtaining these coefficients. Consequently, they ease the comparisons between different ES algorithms.

14. The $(1 + 1)$-ES is the simplest possible evolutionary algorithm. It has not been analyzed in detail on ridge functions. No analytical results have been derived for this algorithm in scope of this work. After obtaining these results, one can conjecture if the $(1 + 1)$-ES can be the most efficient algorithm on unimodal fitness functions.

15. The elitist $(1 + \lambda)$-ES attains smaller stationary values for the progress rate φ and quality gain $\overline{Q}$ than its non-elitist counterpart $(1, \lambda)$-ES on the parabolic ridge (which is unimodal). An interesting task remained is to show for which other functions this result holds. It is important to specify under which general conditions elitism is harmful for the progress in the search space.

16. The quality gain values cannot be used in general to estimate the progress rate values. Conversely, a formula for finding the quality gain values using search space measures φ and φ_R exists for the parabolic ridge. This formula can easily be extended to other ridge functions. Further relations between $\overline{Q}$ and φ are expected to be found in the future.

17. The success probabilities P_{s1} and $P_{s\lambda}$ have not been helpful in general for finding the optimum mutation strength on the ridge functions. Therefore, one cannot use the success probability values in maximizing the progress rate φ. Therefore, other mechanisms are necessary for this purpose.

18. Only the isotropic mutations of the normal distribution have been considered in scope of this work. However, other probability distributions can also be used to generate mutations. Additionally, the mutation models other than the isotropic one can also be applied. These two approaches can be combined with the self-adaptation operator.

The self-adaptation operator. The investigation of the ES algorithms with self-adaptation on ridge functions is attempted in this work, but is not studied in detail. Some results have been obtained for the isotropic mutation model and using the multiplicative self-adaptation rule (2.16). However, they are not included in this work because no conclusive explanations could be derived from the preliminary investigations.

The analysis of the self-adaptation on ridge functions poses an additional difficulty: The quantity r must be considered in the analysis. Since stationary simulations did not yield direct explanations, static experiments are carried out for several σ and r values. The self-adaptation response ψ (see Page 46) has been measured in the experiments.

Two important observations can be given here: Firstly, the progress rate φ also depends on r. Therefore, a given mutation strength σ attains different progress rates at different r values. Moreover, the quality gain values differ from the progress rate values: Therefore, another σ value can yield a larger quality gain, although it has a smaller progress rate value. The quality gain is measured in the fitness space, and the selection operator uses fitness values. Therefore, it is possible that the individuals with better fitness values are selected, although they do not yield a large progress rate. One remedy suggested by Rechenberg is to introduce subpopulations with different σ values and compare their performances in finite time intervals (i.e. hierarchical ES, Page 20). However, there are two difficulties: The isolation time is unknown (and definitely dependent on the fitness function) and the progress rate cannot be obtained in general from fitness measurements. Therefore, there is no guarantee for an improvement of the progress rate if the hierarchical ES is used.

Secondly, the ψ values at stationary $R^{(\infty)}$ distances obtained without the self-adaptation operator are important to determine. If $\psi < 0$, the σ value is expected to decrease in the next generation; otherwise, it is expected to increase if $\psi > 0$. This kind of experiments at $R^{(\infty)}$ values for several σ values will give the stationary value of σ for the self-adaptive ES. For the parabolic ridge, these experiments may explain why the stationary σ value of the self-adaptive ES does not go to infinity. The quality gain curves at these $R^{(\infty)}$ values will also help in a conclusive explanation. The same steps can be used to conjecture why the stationary σ value on the sharp ridge goes to zero or infinity, depending on the d value (see Equation (3.15)). Similarly, the investigations on other fitness functions and on self-adaptation operators other than (2.16) also remain as a matter of future research.

Appendix A

The derivation of $\mathrm{E}\left\{\sum_{i=1}^{N-1}\langle z_i^2\rangle\right\}$

The expression $\mathrm{E}\{\sum_{i=1}^{N-1}\langle z_i^2\rangle\}$ occurs in Point 6.4.2.1. Assuming $\mathrm{E}\{\sum_{i=1}^{N-1}\langle z_i^2\rangle\} \simeq (N-1)\sigma^2$, the relation (6.183) between the expected values $\mathrm{E}\{\langle {r^{(g+1)}}^2\rangle\}$ (6.180) and $\mathrm{E}\{{r^{(g+1)}}^2\}$ (6.181) has been established. This expression stands for the square of the expected average length of the $(N-1)$-dimensional mutation vectors that generated the best μ descendants. In other words, the components orthogonal to the progress direction $\mathbf{v}$ are considered in these $(N-1)$-dimensional vectors. This expected value is derived in the following to show that the previous assumption is correct.

If one denotes the $(N-1)$-dimensional component of the mutation vector $\mathbf{z}$ by $\mathbf{z}'$, this value can be investigated further

$$\begin{aligned}\mathrm{E}\left\{\sum_{i=1}^{N-1}\langle z_i^2\rangle\right\} &= \frac{1}{\mu}\sum_{i=1}^{N-1}\sum_{m=1}^{\mu}\mathrm{E}\left\{z_{i_{m;\lambda}}^2\right\} = \frac{1}{\mu}\sum_{m=1}^{\mu}\sum_{i=1}^{N-1}\mathrm{E}\left\{z_{i_{m;\lambda}}^2\right\} \qquad (\text{A.1})\\ &= \frac{1}{\mu}\sum_{m=1}^{\mu}\mathrm{E}\left\{{\mathbf{z}'}^{\mathrm{T}}_{m;\lambda}\mathbf{z}'_{m;\lambda}\right\} = \frac{1}{\mu}\sum_{m=1}^{\mu}\mathrm{E}\left\{\|\mathbf{z}'_{m;\lambda}\|^2\right\} = \mathrm{E}\left\{\langle\|\mathbf{z}'\|^2\rangle\right\} \quad . \qquad (\text{A.2})\end{aligned}$$

The random variable u will be used to represent the squared length $\|\mathbf{z}'_{m;\lambda}\|^2$. Analogous to (6.184), one obtains

$$\frac{1}{\mu}\sum_{m=1}^{\mu}\mathrm{E}\left\{\|\mathbf{z}'_{m;\lambda}\|^2\right\} = \frac{1}{\mu}\sum_{m=1}^{\mu}\int_0^{\infty} u p_{m;\lambda}(u)\,\mathrm{d}u \quad . \qquad (\text{A.3})$$

The derivation is analogous to the one of $\mathrm{E}\{\langle {r^{(g+1)}}^2\rangle\}$ in Point 6.4.2.2. However, the densities $p(u)$ and $p(Q|u)$ are different. An overview of the derivation will be given below. The similarities to this derivation will be stated explicitly in the following steps. Firstly, the expected value integral (A.3) is determined in more detail. Secondly, the densities $p(u)$ and $p(Q|u)$ are derived. Thirdly, the inner integral $I(Q)$ is solved. Lastly, the outer integration is carried out, and the correctness of the assumption $\mathrm{E}\{\sum_{i=1}^{N-1}\langle z_i^2\rangle\} \simeq (N-1)\sigma^2$ is shown.

The expected value integral. Similar to (6.185)-(6.188), Equation (A.3) is determined. The density $p_{m;\lambda}(u)$

$$p_{m;\lambda}(u) = \frac{\lambda!}{(m-1)!(\lambda-m)!} p(u) P_{a\,m;\lambda}(u) \tag{A.4}$$

and the distribution $P_{a\,m;\lambda}(u)$

$$P_{a\,m;\lambda}(u) = \int_{-\infty}^{\infty} p(Q_{|u}|u) \left[P_1(Q_{|u})\right]^{\lambda-m} \left[1 - P_1(Q_{|u})\right]^{m-1} \mathrm{d}Q_{|u} \tag{A.5}$$

can be inserted back in (A.3). The result reads

$$\begin{aligned} \mathrm{E}\left\{\langle\|\mathbf{z}'\|^2\rangle\right\} &= \frac{1}{\mu}\sum_{m=1}^{\mu}\int_0^{\infty} u \frac{\lambda!}{(m-1)!(\lambda-m)!} p(u) \\ &\times \int_{-\infty}^{\infty} p(Q_{|u}|u) \left[P_1(Q_{|u})\right]^{\lambda-m} \left[1 - P_1(Q_{|u})\right]^{m-1} \mathrm{d}Q_{|u}\, \mathrm{d}u \ . \end{aligned} \tag{A.6}$$

After exchanging the integration order, one obtains

$$\mathrm{E}\left\{\langle\|\mathbf{z}'\|^2\rangle\right\} = \frac{\lambda!}{\mu}\sum_{m=1}^{\mu}\int_{-\infty}^{\infty} \frac{\left[P_1(Q)\right]^{\lambda-m}\left[1-P_1(Q)\right]^{m-1}}{(m-1)!(\lambda-m)!} \underbrace{\int_0^{\infty} u\, p(u)\, p(Q|u)\, \mathrm{d}u}_{I(Q)}\, \mathrm{d}Q \ . \tag{A.7}$$

This will be the starting point of the derivations. The inner integral $I(Q)$ must be derived first. The densities $p(u)$ and $p(Q|u)$ are necessary for this purpose.

The density $p(u)$. This density will be approximated using a normal distribution. Therefore, the expected value and variance of the quantity $\sum_{i=1}^{N-1} z_i^2$ is to be determined. After noticing that $z_i \sim \mathcal{N}(0, \sigma^2)$, using the moments of standard normal distribution $\mathcal{N}(0,1)$ (5.8) and (6.26), one asymptotically $N \to \infty$ obtains

$$\mathrm{E}\left\{\sum_{i=1}^{N-1} z_i^2\right\} = \sum_{i=1}^{N-1}\mathrm{E}\left\{z_i^2\right\} = \sum_{i=1}^{N-1}\overline{z_i^2} = (N-1)\sigma^2 \ , \tag{A.8}$$

$$\begin{aligned} \mathrm{E}\left\{\left(\sum_{i=1}^{N-1} z_i^2\right)^2\right\} &= \sum_{i=1}^{N-1}\mathrm{E}\left\{z_i^4\right\} + \sum_{i=1}^{N-1}\sum_{j\neq i}\mathrm{E}\left\{z_i^2 z_j^2\right\} \\ &= 3(N-1)\sigma^4 + (N-1)(N-2)\sigma^4 \ , \end{aligned} \tag{A.9}$$

$$D^2\left\{\sum_{i=1}^{N-1} z_i^2\right\} = \mathrm{E}\left\{\left(\sum_{i=1}^{N-1} z_i^2\right)^2\right\} - \left[\mathrm{E}\left\{\sum_{i=1}^{N-1} z_i^2\right\}\right]^2 = 2(N-1)\sigma^4 \ . \tag{A.10}$$

Therefore, one gets $u \sim \mathcal{N}((N-1)\sigma^2, 2(N-1)\sigma^4)$

$$p(u) = \frac{1}{\sqrt{2\pi}\sqrt{2(N-1)}\sigma^2} \exp\left[-\frac{1}{2}\left(\frac{u-(N-1)\sigma^2}{\sqrt{2(N-1)}\sigma^2}\right)^2\right] \quad . \tag{A.11}$$

This density was derived in [Bey96c, p. 105] as an intermediate step in obtaining the distribution $P(u)$ (6.141), Page 121. The other density $p(Q|u)$ is derived next, and the evaluation of the integral $I(Q)$ in (A.7) follows.

The density $p(Q|u)$. The derivation of $p(Q|u)$ is a bit lengthy. For this purpose, the definition of $P_1(Q)$ (6.77) will be used. The distribution $P_1(Q)$ defines the distribution of fitness values in the next generation. It has been derived in Point 6.3.2.1 using the distribution of mutations in the progress direction $\mathbf{v}$. Alternative to (6.67), $P_1(Q)$ can also be determined using $p(u)$ and $p(Q|u)$ (denoted as $p(Q_{|u}|u)$)

$$P_1(Q) = \int_{-\infty}^{Q}\int_0^{\infty} p(Q_{|u}|u)\, p(u)\, \mathrm{d}u\, \mathrm{d}Q_{|u} \quad . \tag{A.12}$$

Therefore, $P_1(Q)$ gives the probability of getting a descendant with a local quality function value less than or equal to Q (represented by the outer integral) among all possible mutations (the inner integral). The quantities $P_1(Q)$ and $p(u)$ are known. Therefore, $p(Q_{|u}|u)$ can be determined by solving this integral equation. The integration order is changed first.

$$P_1(Q) = \int_0^{\infty} p(u) \int_{-\infty}^{Q} p(Q_{|u}|u)\, \mathrm{d}Q_{|u}\, \mathrm{d}u \quad . \tag{A.13}$$

The density $p(u)$ (A.11) is inserted after using the substitution

$$t := \frac{u-(N-1)\sigma^2}{\sqrt{2(N-1)}\sigma^2}, \quad \mathrm{d}t = \frac{\mathrm{d}u}{\sqrt{2(N-1)}\sigma^2}, \quad u = \sqrt{2(N-1)}\sigma^2 t + (N-1)\sigma^2 \quad . \tag{A.14}$$

Please note the change in the lower integration limit $u = 0$ to $t = -\infty$ for $N \to \infty$. The inner integral of (A.13) is expected to give a result of the form $\Phi(a \cdot t + b)$. This is not much restrictive since $p(Q|u)$ was approximated by a normal distribution. After these two steps, (A.13) can be solved using (5.11) (see also (6.75) on Page 98)

$$P_1(Q) \stackrel{!}{=} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}t^2}\, \Phi\left[at+b\right] \mathrm{d}t = \Phi\left(\frac{b}{\sqrt{1+a^2}}\right) \quad . \tag{A.15}$$

Using the density $P_1(Q)$ (6.77) and the values of M_Q and S_Q (6.8), one obtains

$$P_1(Q) = \Phi\left(\frac{Q-M_Q}{S_Q}\right) = \Phi\left(\frac{Q+(N-1)d\sigma^2}{\sigma\sqrt{1+(2dr)^2+2d^2(N-1)\sigma^2}}\right) = \Phi\left(\frac{b}{\sqrt{1+a^2}}\right) \quad . \tag{A.16}$$

Therefore, the values of a and b can be identified as

$$a = -\frac{1}{\sqrt{(2dr)^2 + 2d^2(N-1)\sigma^2}}, \qquad b = \frac{Q + (N-1)d\sigma^2}{\sigma\sqrt{(2dr)^2 + 2d^2(N-1)\sigma^2}} \ . \tag{A.17}$$

For a unique solution, a second equation on a and b is needed. These values are chosen to conserve the relationship of $p(Q|u)$ to $p(Q_{|z}|z)$ (see (6.73) and (6.73)). To simplify the notation in the following steps, the variable S

$$\mathrm{S} = \sqrt{(2dr)^2 + 2d^2(N-1)\sigma^2} \tag{A.18}$$

is introduced. The distribution $\Phi(at+b)$ can now be differentiated with respect to Q to obtain $p(Q|t)$

$$p(Q|t) = \frac{\mathrm{d}}{\mathrm{d}Q}\Phi(at+b) = \frac{1}{\sqrt{2\pi}\sigma\mathrm{S}} \exp\left[-\frac{1}{2}\left(\frac{Q - \sigma t + (N-1)d\sigma^2}{\sigma\mathrm{S}}\right)^2\right] \ . \tag{A.19}$$

After substituting u back from (A.14), one obtains the desired density

$$p(Q|u) = \frac{1}{\sqrt{2\pi}\sigma\mathrm{S}} \exp\left[-\frac{1}{2\sigma^2\mathrm{S}^2}\left(Q + (N-1)d\sigma^2 - \frac{u}{\sqrt{2(N-1)}\sigma} + \sqrt{\frac{N-1}{2}}\sigma\right)^2\right] . \tag{A.20}$$

This result can be verified by evaluating (A.13).

The integral $I(Q)$. The evaluation of the integral $I(Q)$ is the next step in solving (A.7). The necessary densities $p(u)$ (A.11) and $p(Q|u)$ (A.20) have already been derived. After inserting them in the $I(Q)$ definition, one obtains

$$\begin{aligned} I(Q) &= \int_0^\infty u\, p(u)\, p(Q|u)\, \mathrm{d}u & \text{(A.21)} \\ &= \frac{1}{2\pi\sqrt{2(N-1)\sigma^2}}\frac{1}{\sigma\mathrm{S}}\int_0^\infty u \exp\left[-\frac{1}{2}\left(\frac{u-(N-1)\sigma^2}{\sqrt{2(N-1)\sigma^2}}\right)^2\right] \\ &\times \exp\left[-\frac{1}{2\sigma^2\mathrm{S}^2}\left(Q + (N-1)d\sigma^2 - \frac{u}{\sqrt{2(N-1)}\sigma} + \sqrt{\frac{N-1}{2}}\sigma\right)^2\right] \mathrm{d}u \ . & \text{(A.22)} \end{aligned}$$

After applying the substitution (A.14), (A.22) becomes

$$I(Q) = \frac{1}{2\pi\sigma\mathrm{S}}\int_{-\infty}^\infty \left(\sqrt{2(N-1)}\sigma^2 t + (N-1)\sigma^2\right) \mathrm{e}^{-\frac{1}{2}t^2}\, \mathrm{e}^{-\frac{1}{2}\left(\frac{Q+(N-1)d\sigma^2-\sigma t}{\sigma\mathrm{S}}\right)^2} \mathrm{d}t \ . \tag{A.23}$$

Please note the change in the lower integration boundary. Using the definitions of a and b in (A.17), and the integral expressions (5.9) and (5.10), respectively, the solution of (A.23) reads

$$I(Q) = \frac{1}{\sqrt{2\pi}\sigma \mathrm{S}} \left[\sqrt{2(N-1)}\sigma^2 \frac{-ab}{1+a^2} + (N-1)\sigma^2 \right] \frac{1}{\sqrt{1+a^2}} \mathrm{e}^{-\frac{1}{2}\frac{b^2}{1+a^2}} \ . \tag{A.24}$$

The values of a and b are substituted back by considering the definitions of M_Q and S_Q (6.8). Please note that

$$1 + a^2 = 1 + \frac{1}{\mathrm{S}^2} = \frac{1+\mathrm{S}^2}{\mathrm{S}^2} = \frac{S_Q^2}{\sigma^2 \mathrm{S}^2} \ , \tag{A.25}$$

$$\frac{b^2}{1+a^2} = \frac{\sigma^2 \mathrm{S}^2}{S_Q^2} \frac{\left(Q + (N-1)d\sigma^2\right)^2}{\sigma^2 \mathrm{S}^2} = \left(\frac{Q - M_Q}{S_Q}\right)^2 \ . \tag{A.26}$$

Considering (A.17), (A.25), and (A.26), (A.24) becomes

$$I(Q) = \frac{1}{\sqrt{2\pi} S_Q} \left[\sqrt{2(N-1)}\sigma^3 \frac{Q - M_Q}{S_Q^2} + (N-1)\sigma^2 \right] \mathrm{e}^{-\frac{1}{2}\left(\frac{Q-M_Q}{S_Q}\right)^2} \ . \tag{A.27}$$

The outer integral and its solution. In the last step, the solution of $I(Q)$ (A.27) is inserted back into (A.7). Considering (A.16), and the substitution

$$s = \frac{Q - M_Q}{S_Q}, \qquad \mathrm{d}s = \frac{\mathrm{d}Q}{S_Q} \ , \tag{A.28}$$

(A.7) becomes

$$\begin{aligned} \mathrm{E}\left\{ \langle \|\mathbf{z}'\|^2 \rangle \right\} &= \frac{\lambda!}{\sqrt{2\pi}\mu} \int_{-\infty}^{\infty} \left[\sqrt{2(N-1)}\sigma^3 \frac{s}{S_Q} + (N-1)\sigma^2 \right] \mathrm{e}^{-\frac{1}{2}s^2} \\ &\quad \times \sum_{m=1}^{\mu} \frac{[\Phi(s)]^{\lambda-m} [1-\Phi(s)]^{m-1}}{(m-1)!(\lambda-m)!} \mathrm{d}s \ . \end{aligned} \tag{A.29}$$

One observes a remarkable similarity between (A.29) and (6.191): Only the contents of the braces differ from each other. Therefore, the rest of the derivation is analogous to (6.192)-(6.196), and will not be repeated here. The result (6.196) on Page 134 can be adapted to (A.29): If the brace is abstracted as $[As + B]$, the result reads for both cases

$$\mathrm{E}\{.\} = A\, e_{\mu,\lambda}^{1,0} + B\, e_{\mu,\lambda}^{0,0} = A c_{\mu/\mu,\lambda} + B \ . \tag{A.30}$$

The correspondence of $e_{\mu,\lambda}^{1,0}$ to $c_{\mu/\mu,\lambda}$ and $e_{\mu,\lambda}^{0,0}$ to 1 has been explained on Page 134. After inserting the values of A and B from (A.29), and the value of S_Q from (6.8), the final result reads

$$\boxed{\mathrm{E}\left\{ \sum_{i=1}^{N-1} \langle z_i^2 \rangle \right\} = (N-1)\sigma^2 + \sqrt{\frac{2(N-1)}{1 + (2dr)^2 + 2d^2(N-1)\sigma^2}}\sigma^2 c_{\mu/\mu,\lambda}} \ . \tag{A.31}$$

This result was obtained for $N \to \infty$ and after approximating the distributions of the Q variates by normal distributions. In the derivation, the lower integration limit has been extended to $-\infty$ to obtain (A.23).

The second term of (A.31) can be neglected in the static case as $N \to \infty$. For the stationary case, the second term becomes $\mathcal{O}(N^{-1/2})$, since $r = r^{(g)}$ is of $\mathcal{O}(N)$ in this case. Consequently, $\mathrm{E}\{\langle \|\mathbf{z}'\|^2 \rangle\} = \mathrm{E}\{\sum_{i=1}^{N-1} \langle z_i^2 \rangle\}$ is of $\mathcal{O}(N\sigma^2)$. As a result, one obtains $\mathrm{E}\{\sum_{i=1}^{N-1} \langle z_i^2 \rangle\} \simeq (N-1)\sigma^2$. This simpler result can be used in the calculation of $R^{(\infty)}$. Otherwise, one has to use in (6.198) the expression (A.31) instead of $(N-1)\sigma^2$ in the last additive term for a more accurate state equation.

Bibliography

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

[Ala94] J. T. Alander. An Indexed Bibliography of Genetic Algorithms: Years 1957-1993. Vaasa (Finland), 1994.

[AS84] M. Abramowitz and I. A. Stegun. *Pocketbook of Mathematical Functions.* Verlag Harri Deutsch, Thun, 1984.

[Bäc96] T. Bäck. *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, New York, 1996.

[Bey93] H.-G. Beyer. Toward a Theory of Evolution Strategies: Some Asymptotical Results from the $(1 \overset{+}{,} \lambda)$–Theory. *Evolutionary Computation*, 1(2):165–188, 1993.

[Bey94] H.-G. Beyer. Towards a Theory of 'Evolution Strategies'. Progress Rates and Quality Gain for $(1 \overset{+}{,} \lambda)$–Strategies on (Nearly) Arbitrary Fitness Functions. In Davidor et al. [DSM94], pages 58–67.

[Bey95a] H.-G. Beyer. Toward a Theory of Evolution Strategies: On the Benefits of Sex–the $(\mu/\mu, \lambda)$ Theory. *Evolutionary Computation*, 3(1):81–111, 1995.

[Bey95b] H.-G. Beyer. Toward a Theory of Evolution Strategies: The (μ, λ)–Theory. *Evolutionary Computation*, 2(4):381–407, 1995.

[Bey96a] H.-G. Beyer. On the Asymptotic Behavior of Multirecombinant Evolution Strategies. In Voigt et al. [VERS96], pages 122–133.

[Bey96b] H.-G. Beyer. Toward a Theory of Evolution Strategies: Self-Adaptation. *Evolutionary Computation*, 3(3):311–347, 1996.

[Bey96c] H.-G. Beyer. *Zur Analyse der Evolutionsstrategien.* Habilitationsschrift. University of Dortmund, Department of Computer Science, 1996. In German.

[Bey97] H.-G. Beyer. An Alternative Explanation for the Manner in which Genetic Algorithms operate. *BioSystems*, 41(1):1–15, 1997.

[Bey98] H.-G. Beyer. On the "Explorative Power" of ES/EP-like Algorithms. In V.W. Porto, N. Saravanan, D. Waagen, and A.E. Eiben, editors, *Evolutionary Programming VII: Proceedings of the Seventh Annual Conference on Evolutionary Programming*, pages 323–334, Heidelberg, 1998. Springer-Verlag.

[BFM97] T. Bäck, D. B. Fogel, and Z. Michalewicz, editors. *Handbook of Evolutionary Computation.* Oxford University Press, New York, and Institute of Physics Publishing, Bristol, 1997.

[BNKF98] W. Banzhaf, P. Nordin, R. E. Keller, and F. D. Francone. *Genetic Programming: An Introduction.* Morgan Kaufmann Publishers, San Francisco, 1998.

[BS79] I. N. Bronstein and K. A. Semendjajew. *Taschenbuch der Mathematik.* B.G. Teubner, Leipzig, 1979.

[BS91] I. N. Bronstein and K. A. Semendjajew. *Taschenbuch der Mathematik.* B.G. Teubner, Stuttgart, 1991.

[BS93] T. Bäck and H.-P. Schwefel. An Overview of Evolutionary Algorithms for Parameter Optimization. *Evolutionary Computation*, 1(1):1–23, 1993.

[Deb98] K. Deb. An Efficient Constraint Handling Method for Genetic Algorithms. *Computer Methods in Applied Mechanics and Engineering*, 1998. in print.

[DSM94] Y. Davidor, H.-P. Schwefel, and R. Männer, editors. *Parallel Problem Solving from Nature — PPSN III, International Conference on Evolutionary Computation*, volume 866 of *Lecture Notes in Computer Science.* Springer, Berlin, 1994.

[EBSS98] A. E. Eiben, Th. Bäck, M. Schoenauer, and H.-P. Schwefel, editors. *Parallel Problem Solving from Nature – PPSN V, International Conference on Evolutionary Computation*, volume 1498 of *Lecture Notes in Computer Science.* Springer, Berlin, 1998.

[ERR94] A. E. Eiben, P.-E. Raué, and Zs. Ruttkay. Genetic algorithms with multi-parent recombination. In Davidor et al. [DSM94], pages 78–87.

[Esh95] L. Eshelman, editor. *Genetic Algorithms: Proceedings of the 6th International Conference.* Morgan Kaufmann Publishers, San Francisco, CA, 1995.

[Fel71] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume 2. John Wiley & Sons, New York, 2nd edition, 1971.

[Fis76] M. Fisz. *Wahrscheinlichkeitsrechnung und mathematische Statistik.* VEB, Dt. Verlag der Wissenschaften, Berlin, 8. edition, 1976.

[Fog92] D. B. Fogel. *Evolving Artificial Intelligence.* Ph.D. Thesis, University of California, San Diego, 1992.

[Fog95] D. B. Fogel. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence.* IEEE Press, Piscataway, NJ, 1995.

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

[GL97] F. W. Glover and M. Laguna. *Tabu Search.* Kluwer Academic Publishers, Boston, 1997. See `http://spot.colorado.EDU/~glover/publications.html` for further publications.

[Glo86] F. W. Glover. Future Paths for Integer Programming and Links to Artificial Intelligence. *Computers and Operations Research*, 13(5):533–549, 1986.

[Glo89a] F. W. Glover. Tabu Search – Part I. *ORSA Journal on Computing*, 1(3):190–206, 1989.

[Glo89b] F. W. Glover. TABU Search: a Tutorial. University of Colorado, Boulder, CO, Nov 1989.

[Gol89] D. E. Goldberg. *Genetic Algorithms in Search, Optimization & Machine Learning.* Addison-Wesley, Reading, 1989.

[GTdW91] F. W. Glover, E. Taillard, and D. de Werra. *A User's Guide to Tabu Search.* University of Colorado and Swiss Federal Institute of Technology, Nov 1991.

[Her92] M. Herdy. Reproductive Isolation as Strategy Parameter in Hierarchically Organized Evolution Strategies. In Männer and Manderick [MM92], pages 207–217.

[HM79] C.-L. Hwang and A. S. M. Masud. *Multiple Objective Decision Making - Methods and Applications.* Springer, Berlin, 1979.

[Hol75] J. H. Holland. *Adaptation in Natural and Artificial Systems.* Univ. of Michigan, NN, 1975.

[Kap96] C. Kappler. Are Evolutionary Algorithms Improved by Large Mutations? In Voigt et al. [VERS96], pages 346–355.

[KGV83] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, 1983.

[Koz94] J. R. Koza. *Genetic Programming II.* MIT Press, Cambridge, MA, 1994.

[KS70] J. Klockgether and H.-P. Schwefel. Two-phase nozzle and hollow core jet experiments. In D. G. Elliott, editor, *Proc. Eleventh Symp. Engineering Aspects of Magnetohydrodynamics*, pages 141–148, Pasadena CA, March 24-26 1970. California Institute of Technology.

[Kur91] F. Kursawe. A Variant of Evolution Strategies for Vector Optimization. In Schwefel and Männer [SM91], pages 193–197.

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

[MMMC95] F. Morán, A. Moreno, J. J. Merelo, and P. Chacón, editors. *Advances in Artificial Life. Third International Conference on Artificial Life*, volume 929 of *Lecture Notes in Artificial Intelligence*. Springer, Berlin, June 1995.

[Müc89] A. Mück. Einfluß verschiedener Wahrscheinlichkeitsverteilungen auf das Konvergenzverhalten von Evolutionsstrategien. Diplomarbeit (MS Thesis), University of Dortmund, Department of Computer Science, August 1989.

[Nor97] P. Nordin. *Evolutionary Program Induction of Binary Machine Code and its Application.* Krehl Verlag, Münster, 1997.

[OBS97] A. I. Oyman, H.-G. Beyer, and H.-P. Schwefel. Analysis of a Simple ES on the "Parabolic Ridge". Technical Report SyS–2/97, University of Dortmund, Department of Computer Science, Systems Analysis Research Group, August 1997.

[OBS98a] A. I. Oyman, H.-G. Beyer, and H.-P. Schwefel. Convergence Behavior of the $(1 \overset{+}{,} \lambda)$ Evolution Strategy on the Ridge Functions. Technical Report SyS–1/98, University of Dortmund, Department of Computer Science, Systems Analysis Research Group, February 1998.

[OBS98b] A. I. Oyman, H.-G. Beyer, and H.-P. Schwefel. Convergence Behavior of the $(1 \overset{+}{,} \lambda)$ Evolution Strategy on the Ridge Functions. *Mathware & Soft Computing*, 1998. `submitted: April 1998`, `accepted: November 1998`.

[OBS98c] A. I. Oyman, H.-G. Beyer, and H.-P. Schwefel. Where Elitists Start Limping: Evolution Strategies at Ridge Functions. In Eiben et al. [EBSS98], pages 34–43.

[OBS99] A. I. Oyman, H.-G. Beyer, and H.-P. Schwefel. Analysis of a Simple ES on the "Parabolic Ridge". *Evolutionary Computation*, 1999. in print.

[Ost97] A. Ostermeier. *Schrittweitenadaptation in der Evolutionsstrategie mit einem entstochastisierten Ansatz.* Dr.-Ing. Dissertation (PhD Thesis), Technical University of Berlin, Department of Process Engineering, 1997.

[Ott93] K. Ott. Einfluß stochastischer Störungen auf das Konvergenzverhalten von Evolutionsstrategien. Diplomarbeit (MS Thesis), University of Dortmund, Department of Computer Science, May 1993.

[Pin97] A. Pinkus. Approximating by Ridge Functions. In A. Le Mehaute, C. Rabut, and L. L. Schumaker, editors, *Surface Fitting and Multiresolution Methods*, pages 279–292. Vanderbilt University Press, Nashville, 1997.

[PTVF92] W.H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, editors. *Numerical Recipes in C, The Art of Scientific Computing.* Cambridge University Press, Cambridge, 2nd edition, 1992.

[Rec65] I. Rechenberg. Cybernetic solution path of an experimental problem. Library translation 1122, Royal Aircraft Establishment, Farnborough, 1965.

[Rec71] I. Rechenberg. *Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.* Dr.-Ing. Dissertation (PhD Thesis), Technical University of Berlin, Department of Process Engineering, 1971.

[Rec73] I. Rechenberg. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.* Verlag Frommann-Holzboog, Stuttgart-Bad Cannstatt, 1973. ISBN: 3-7728-0373-3, in German.

[Rec78] I. Rechenberg. Evolutionstrategien. In B. Schneider and U. Ranft, editors, *Simulationsmethoden in der Medizin und Biologie*, pages 83–114. Springer, Berlin, 1978.

[Rec94] I. Rechenberg. *Evolutionsstrategie'94*. Band 1, Werkstatt Bionik und Evolutionstechnik. Frommann–Holzboog, Stuttgart, 1994. ISBN: 3-7728-1642-8, in German.

[Roh76] Vijay K. Rohatgi. *An Introduction to Probability Theory and Mathematical Statistics*. Series in probability and mathematical statistics. Wiley, New York, 1976.

[Rud94] G. Rudolph. Massively Parallel Simulated Annealing and Its Relation to Evolutionary Algorithms. *Evolutionary Computation*, 1(4):361–383, 1994.

[Rud97] G. Rudolph. *Convergence Properties of Evolutionary Algorithms*. Verlag Dr. Kovač, Hamburg, 1997.

[Sal96] R. Salomon. Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions. A survey of some theoretical and practical aspects of genetic algorithms. *BioSystems*, 39(3):263–278, 1996.

[Sch65] H.-P. Schwefel. *Kybernetische Evolution als Strategie der experimentellen Forschung in der Strömungstechnik*. Dipl.-Ing. Thesis, Technical University of Berlin, Hermann Föttinger–Institute for Hydrodynamics, March 1965.

[Sch75] H.-P. Schwefel. *Evolutionsstrategie und numerische Optimierung*. Dr.-Ing. Dissertation (Ph.D. Thesis), Technical University of Berlin, Department of Process Engineering, 1975.

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

[Sch81] H.-P. Schwefel. *Numerical optimization of computer models*. Wiley, Great Britain, 1981.

[Sch95] H.-P. Schwefel. *Evolution and Optimum Seeking*. Sixth–Generation Computer Technology Series. Wiley, New York, 1995. ISBN: 0-471-57148-2.

[Sch97] H.-P. Schwefel. *Personal communication*, 1997.

[SM91] H.-P. Schwefel and R. Männer, editors. *Parallel Problem Solving from Nature — Proceedings of the 1st PPSN Workshop*, volume 496 of *Lecture Notes in Computer Science*. Springer, Berlin, Oct. 1-3, 1990 1991.

[SR95] H.-P. Schwefel and G. Rudolph. Contemporary Evolution Strategies. In F. Morán, A. Moreno, J. J. Merelo, and P. Chacón, editors, *Advances in Artificial Life. Third International Conference on Artificial Life*, pages 893–907. Springer, Berlin, 1995.

[SV98] D. Schlierkamp-Voosen. *Populationsbasierte Wettbewerbsmodelle zur Strategieanpassung in Evolutionären Algorithmen*. PhD Thesis, University of Dortmund, Department of Computer Science, 1998.

[TF96] G. B. Thomas and R. L. Finney. *Calculus and Analytic Geometry*. Addison-Wesley Longman, Inc., Reading, MA, 9th edition, 1996. `http://hepg.awl.com/AWBookCatalog/Book/book.asp?BOOK_ID=4`.

[VERS96] H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, editors. *Parallel Problem Solving from Nature – PPSN IV, International Conference on Evolutionary Computation*, volume 1141 of *Lecture Notes in Computer Science*. Springer, Berlin, 1996.

[VMC95] H.-M. Voigt, H. Mühlenbein, and D. Cvetković. Fuzzy recombination for the Breeder Genetic Algorithm. In Eshelman [Esh95], pages 104–111.

[Yer96] D. Yerkes. *Webster's Encyclopedic Unabridged Dictionary of the English Language*. Gramercy Books, Random House Value Publishing, Inc., New Jersey, 1996.

Index

Curriculum Vitae

Name:	Ahmet İrfan Oyman
Date of birth:	4.12.1969
Place of birth:	Konya, Turkey
Marital status:	Married

1975-1980	Primary education
1980	Entrance examination for the secondary education
1980-1987	Secondary education in “İstanbul Lisesi”
1987	Graduation with an exceptional degree qualifying for a DAAD scholarship
1987	The University Placement Test of the Turkish Higher Education Council
1987-1992	Undergraduate education in the Department of Computer Engineering at the Boğaziçi University, İstanbul
1992	Bachelor of Science degree, with honors
1992	B.Sc. thesis on “Neocognitron Artificial Neural Network” (programming)
1992-1995	Graduate education in the same department
1992-1995	Research assistant in the same department
1995	Master of Science degree
1995	M.Sc. thesis, “Feature Selection Using Genetic Algorithms”
1995	Starting doctoral studies with a DAAD scholarship in the Department of Computer Science at the University of Dortmund

Languages:	Fluent in German, English, and Turkish
Research interests:	Evolutionary algorithms, experimental and theoretical analysis of the convergence behavior, investigation of the operating mechanisms of evolutionary operators, probabilistic optimization algorithms, systems analysis.