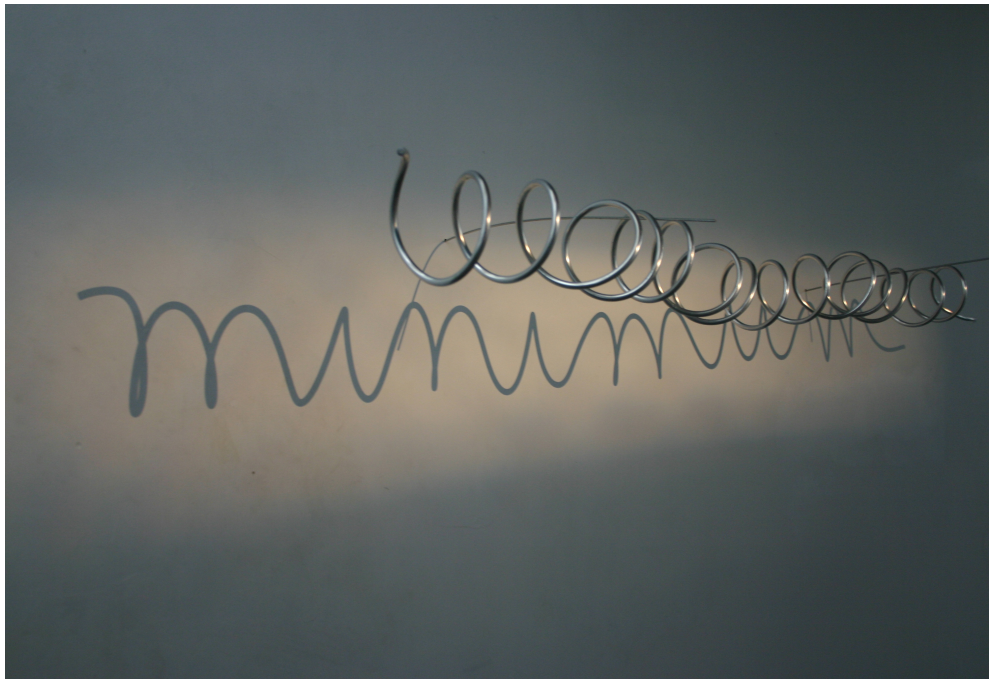
Diss. ETH No. 19438 2010

# Black-box Landscapes: Characterization, Optimization, Sampling, and Application to Geometric Configuration Problems



**Christian L. Müller**

Diss. ETH No. 19438

# Black-box Landscapes: Characterization, Optimization, Sampling, and Application to Geometric Configuration Problems

*A dissertation submitted to*
**ETH Zürich**

*for the degree of*
Doctor of Sciences

*presented by*
**Christian L. Müller**
M.Sc., Uppsala Universitet
Dipl. Inf./Bioinf., University of Tübingen

*born* 14 april 1979
*citizen of* Schesslitz - Germany

*accepted on the recommendation of*
Prof. Dr. Ivo F. Sbalzarini, examiner
Dr. Nikolaus Hansen, co-examiner
Dr. Philippe H. Hünenberger, co-examiner
Prof. Dr. Emo Welzl, co-examiner
Dr. Bojan Žagrović, co-examiner

2010

"Dicebat Bernardus Carnotensis nos esse *quasi nanos gigantum umeris insidentes*, ut possimus plura eis et remotiora videre, non utique proprii visus acumine, aut eminentia corporis, sed quia in altum subvehimur et extollimur magnitudine gigantea."

John of Salisbury, Metalogicon 3,4,46-50

Cover photo © Fred Eerdekens
*minimum*
2004
aluminium
168 x 16 x 18 cm

# Abstract

In many areas of science and engineering researchers consider systems that can be solely examined by their input and output characteristics without any knowledge of their internal workings. Such black-box systems are the topic of the present thesis. In many practical cases, a black box comprises a complex mathematical model, a computer simulation, a real-world experiment, or a combination of any of these. In this thesis we take an interdisciplinary approach to the characterization, optimization, and sampling of black-box systems. We focus on systems with high-dimensional real-valued input variables and output patterns that can be transformed by some function into a scalar real-valued quantity. Throughout this thesis we conceptualize the black-box system as a *landscape*. Inspired by our shared visual experience of natural terrains and sceneries, we consider the real-valued input variables as a high-dimensional landscape domain. Neighborhood or nearness in this landscape domain must be provided by a suitable distance metric. We interpret the scalar output quantity as a height or elevation over the landscape domain. The landscape metaphor encourages a characterization of black-box systems in terms of topographical features, such as valleys, ridges, mountain peaks, and plateaus. In order to underline that we view black-box systems as high-dimensional, complex landscapes we introduce the notion of the *black-box landscape*. After a general review of the landscape paradigm, spanning the disciplines of biology, physics, chemistry, and optimization, we present a number of statistical landscape descriptors that probe different properties of black-box landscapes. The core of the thesis is concerned with black-box optimization. We improve the performance of the arguably best state-of-the-art optimizer, the Covariance Matrix Adaptation Evolution Strategy (CMA-ES), in various aspects. The general performance is increased by considering quasi-random instead of pseudo-random sampling. For multi-funnel landscape topologies we introduce parallel CMA-ES schemes that can outperform standard CMA-ES. We also revisit Gaussian Adaptation, an optimization and sampling scheme that has been largely ignored in the black-box optimization community. Our improved Gaussian Adaptation scheme shows remarkable performance on the considered benchmarks and ranks among the best known black-box optimizers. An important conceptual result is that we can provide an explicit link between black-box optimization and black-box (or indirect) sampling through Gaussian Adaptation. We show that the same idea of adaptation has emerged in these disparate fields, and we argue that a unifying framework for sampling and optimization might constitute an important contribution. We further consider geometric configurations in two different contexts: Geometry optimization problems of atomic clusters are proposed as novel benchmarks for black-box optimization. We design a balanced set of problems that should be included in future black-box optimization benchmarks. We also revisit the configuration space of chain molecules with respect to a certain distance measure, the Root Mean Square Deviation (RMSD) after optimal superposition. Because RMSD is the most important distance metric

in structural biology, we quantify the neighborhood structure that is induced by the RMSD for the Random Walk polymer model. Based on numerical results from black-box optimization runs, we are also able to formulate a conjecture about an upper bound of the RMSD between any two Random Walks of arbitrary length. In the course of the thesis, two software libraries for black-box sampling and optimization, GaALib and pCMALib, have been developed that might prove valuable for the scientific community.

# Zusammenfassung

Viele Systeme und Modelle in Wissenschaft und Technik können aufgrund ihres hohen Komplexitätsgrades nur noch bezüglich ihrer Ein- und Ausgangseigenschaften beschrieben und analysiert werden. In vielen Fällen ist detailliertes Wissen über interne Systemabläufe und -zusammenhänge nicht mehr zugänglich. Solche, so genannte Black-Box-Systeme sind das Thema der vorliegenden Arbeit. Komplexe, mathematische Modelle, Computersimulationen, aufwendige Laborexperimente sowie beliebige Kombinationen von Labor- und Computerexperimenten lassen sich als Black-Box-Systeme modellieren. Die vorliegende Arbeit präsentiert einen interdisziplinären Ansatz zur Charakterisierung, Optimierung und zum randomisierten Abtasten solcher Systeme, wobei das Hauptaugenmerk auf Modellen mit hochdimensionalen, reellwertigen Eingangsgrössen und skalaren, reellwertige Ausgangsgrössen liegt. Eine Besonderheit dieser Arbeit liegt in der Betrachtungsweise eines Black-Box-Systems als hochdimensionale, abstrakte Landschaft: die Black-Box-Landschaft. Diese Metapher ermöglicht einen anschaulichen, topographisch inspirierten Zugang zur Systemanalyse. Die reelle Eingangsgrössen definieren darin einen hochdimensionalen Raum, die skalare Ausgangsgrösse eine Höhenangabe für jeden Punkt im Raum. Nachbarschaft oder Nähe in einer solchen Landschaft wird durch ein geeignetes Abstandsmass, z.B. die Euklidische Distanz, bestimmt. Eine Charakterisierung von Black-Box-Systemen kann nun mit Hilfe topographischer Begriffe, wie zum Beispiel Täler, Grate, Gipfel oder Plateaus, erfolgen. Das Landschaftsparadigma ist ein zentraler Bestandteil der Molekularphysik, der Evolutionsbiologie sowie der kombinatorischen Optimierung. Nach einer Analyse der wichtigsten Arbeiten aus diesen Wissenschaftsgebieten stellen wir eine Reihe von statistischen Verfahren vor, mit denen sich verschiedene Merkmale von Black-Box-Landschaften beschreiben lassen. Ein wichtiger Bestandteil dieser Arbeit ist die effiziente Optimierung von Black-Box-Systemen. Wir verbessern verschiedene Komponenten einer der besten Black-Box-Optimierungsmethoden, der Evolutionsstrategie mit Kovarianzmatrixanpassung (Covariance Matrix Adaptation Evolution Strategy, CMA-ES). Das Abtastverfahren der Strategie wird durch die Verwendung von Quasi-Zufallszahlen anstelle von Pseudozufallszahlen für die Generierung von Stichproben gesteigert. Für die effiziente Exploration von Black-box-Landschaften, die mehrere tiefe, trichterförmige Täler aufweisen, d.h. für Systeme, die weit auseinander liegende Bereiche im Eingangsraum besitzen, die ähnlich optimale Ausgangsgrössen liefern, führen wir parallele CMA-ES-Suchmethoden ein. Diese Strategien können die Effizenz im Vergleichen zu sequentiellen Varianten der CMA-ES für bestimmte Modellprobleme steigern. Darüber hinaus greifen wir die Methode der Gauss'schen Anpassung (Gaussian Adaptation, GaA) wieder auf, einem Optimierungs- und Abtastverfahren, dem bislang in der Wissenschaftsgemeinde wenig Beachtung geschenkt wurde. Wir verbessern das ursprüngliche Verfahren und demonstrieren seine Effektivität auf einer grossen Klasse von Testproblemen. Darüber hinaus weisen wir nach, dass die Methode

der Gauss'schen Anpassung die Möglichkeit eröffnet, die Optimierung und Stichprobennahme für Black-Box-Systeme zu vereinheitlichen. Geometrische Konfigurationsprobleme werden in dieser Arbeit in zweierlei Hinsicht berücksichtigt. Zum einen entwerfen wir ein neuartiges Set von geometrischen Optimierungsproblemen, das auf der Energieminimierung atomarer Cluster beruht. Wir analysieren die Topographie der resultierenden Energielandschaften und zeigen, dass die behandelten Probleminstanzen als anspruchsvolle Benchmarks für Black-Box-Optimierungsmethoden dienen können. Zum zweiten beschäftigen wir uns mit dem Konfigurationsraum von Kettenmolekülen in Bezug auf eine bestimmte Distanz, die mittlere quadratische Abweichung (Root Mean Square deviation, RMSD) nach optimaler Superposition. Da RMSD die wichtigste Distanzmetrik der Strukturbiologie darstellt, quantifizieren wir die von ihr induzierte Nachbarschaftstruktur für das einfachste Polymermodell, das Random-Walk-Modell. Darüber hinaus ermöglicht eine Kombination von numerischen Black-Box-Optimierungsexperimenten und geometrischen Überlegungen das Aufstellen einer Vermutung über eine obere Schranke für den RMSD zwischen zwei beliebigen Random-Walks beliebiger Länge. Im Laufe der Arbeit wurden des weiteren zwei öffentlich zugänglich Softwarebibliotheken für Black-Box-Optimierung und Black-box-Stichprobennahme entwickelt, GaALib und pCMALib, die der Wissenschaftsgemeinde möglicherweise von Nutzen sein können.

# Acknowledgements

> "Thank you, come again."
> Apu Nahasapeemapetilon, in: The Simpsons, Sweet Seymour Skinner's Baadasssss Song, Episode no. 100, 1994

First and foremost, I would like to thank my supervisor Ivo Sbalzarini. His great support and trustfulness were invaluable. Ivo was responsive whenever help was needed. His patience was exemplary when research got stuck in dead ends or took much longer than expected. Ivo gave me the opportunity to follow my own scientific ideas within the open, undogmatic environment of the MOSAIC group and encouraged me to present my work on many different conferences, seminars, and workshops. Thank you, Ivo!
When Ivo started the MOSAIC group (or better said, the good old Computational Biophysics Lab) with Jo and me as PhD students, there existed only the vision of an interdisciplinary group doing research at the interface of biology, physics, and computer science. With every new group member the diversity of research topics, scientific skills, and scientific perspectives grew, and the vision soon became reality. Having been a member of this group from the start was an invaluable experience for me.
At the same time, doing interdisciplinary research in such a young group was extremely challenging. A great deal of endurance and serenity was needed to overcome the difficulties along the way. I was, however, very fortunate to meet and interact with a number of great personalities during the past years who supported me in various ways. Their help was invaluable for my research. First of all, I'd like to thank all past and present members of the MOSAIC group. Jo's careful skepticism nicely balanced my own sometimes exuberant enthusiasm. Birte's gentleness and thoughtfulness made her a very enjoyable colleague over the past four years. Since the inspiring MedILS summer school on interdisciplinary research in 2006 I have found a true companion and later an amicable office mate in Greg. He was a constant source of inspiration and knowledge about statistics, evolutionary biology, and science in general. Without him, this work would not have been the same. Special thanks go to Janick for countless discussions and encouraging remarks on my works on optimization and MCMC. I also thank Rajesh for initial help with the RMSD upper bound conjecture and Sylvain for many discussions on clusters and pair potentials. Since our joint stay at MedILS as allies in protein research I'm grateful to have Omar as kind and supportive colleague. I'm also thankful for many vivid discussion with Nelido. He truly taught me the physicist's mindset.
Further thanks go to my non-MOSAICIAN colleagues Tilman Lange, Marco Terzer, Thomas Fuchs, Manfred Claassen, and Peter Bayer for many discussions about almost everything,

including science.
During my time as PhD student, I had the opportunity to supervise a gang of very talented students, all of whom contributed with thoughts, ideas, and computer code to the present thesis. I'm especially grateful to Johannes Lederer, Benedikt Baumgartner, Yannick Misteli, Christian Fiegl, Georg Ofenbeck, and Markus König for their great dedication. Benedikt, Georg, and Markus deserve special mentioning for their contributions to pCMALib.
One of the initial topics of my PhD was low-dimensional description of proteins. Prof. Andrew Pohorille is greatly acknowledged for this idea. Although research in this direction turned out to be infeasible for a single PhD, it nonetheless triggered my interests in proteins. While attending a lecture on computer simulations in biology at ETH, I started working on a student project about structural motifs, supervised by Bojan Žagrović. I've never met a person like Bojan before who was so enthusiastic about structural motifs in proteins, most prominently, helices (see cover photo). I'm thankful for many scientific and non-scientific discussions and I'm honored to have him as one of my co-examiners. Philippe Hünenberger soon joined the project on structural motifs and complemented our computational results by a great deal of analytic work. I'm grateful to Phil for many discussions, and I'm glad and honored to have him as a co-examiner as well.
My investigations on random walks have been partially conducted at MedILS in Split where I was fortunate to meet a gang of young and talented researchers. Special thanks go to Anita Krisko for many valuable discussions about proteins and beyond.
I am grateful to Scott Woodley for organizing the CCP5 2009 Annual Meeting on Structure Prediction in chemistry and accepting me as a presenter and workshop participant. This workshop allowed me to meet the experts in the field of energy landscapes. Special thanks go to Prof. Roy Johnston, Kathleen Steinhöfel, and Andreas Albrecht for interesting discussions during this meeting and in Birmingham soon after. I'm also grateful to Prof. Christian Schön for inviting me to the 2010 Energy Landscape workshop held in Chemnitz. I thank all participants for the very lively discussions, above all, Prof. Peter Salamon for his thoughtful doubts and suggestions regarding my work, and Fréderic Cazals for continuing discussions about computational geometry and proteins.
When I started my PhD in Zurich, I was glad to have Nikolaus Hansen in our institute. His work on CMA-ES and his scientific mindset were a true inspiration for my own investigations in the field of black-box optimization. I'm glad and feel honored that he agreed on being a member of my PhD committee. I'm also grateful to Per-Kristian Lehre and Pietro Oliveto for organizing the 2009 meeting on the theory of randomized search heuristics. Special thanks go to Anne Auger for many discussions on CMA-ES and for inviting me to the 2010 Dagstuhl seminar on the theory of evolutionary computation. This was a truly exciting meeting.
My recent interest in theoretical aspects of black-box optimization has also been triggered by Bernd Gärtner. Thanks to him, I could become a member of the consortium on computational geometric learning. I'm grateful to Bernd for many vivid discussions in the past months. I'm also grateful to Prof. Emo Welzl who agreed to be a co-examiner of my thesis.
My outmost thanks go to my parents, my sister Silke, and Guido for their constant support and encouragement throughout the past years. Finally, thank you, Sonja. I know it was tough.

Zürich in November 2010
Christian

# Contents

# 1 Introduction

> "Science? What's science ever done for us?"
> Moe Szyslak in: The Simpsons, Lisa the Skeptic, Episode no. 186, 1997

In many areas of science, engineering, and economics researchers and decision makers are faced with the task of characterizing and optimizing a system that can solely be examined through its input and output characteristics, without any knowledge of its internal workings. Such systems are generally referred to as *black-box* systems. Fig. 1.1 sketches the black-box concept.

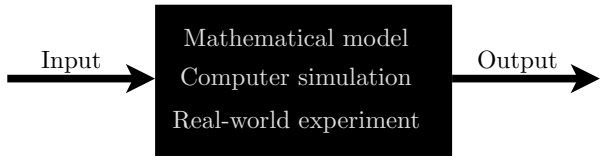



Figure 1.1: Sketch of the black-box paradigm. An input is provided to a black box. The black box can comprise a mathematical model, a computer experiment, a real-world experiment, or a mixture of any of these components. The output is the only observable of the system.

In many practical cases, a black box comprises a complex mathematical model, a computer simulation, a real-world experiment, or a combination of any of these. In practice, the assumption of complete lack of knowledge about the internal characteristics is unrealistic. The black-box model rather expresses our inability to comprehend the complex interactions and causal connections present in the system. Thorough understanding of a black-box system is

usually provided by investigating its transfer characteristics of the black-box. Such analysis comprises inference about the relationship between input and corresponding output. Major objectives in black-box analysis are (i) the quantification of how variation of the output can be explained by the variation (of subsets) of input patterns (*black-box characterization*), (ii) retrieval of a specific element among the set of all possible inputs that is optimal with respect to some properties of the output (*black-box optimization*), and (iii) generation of input patterns according to some probability distribution (*black-box sampling*).

In this thesis we take an interdisciplinary approach to the characterization, optimization, and sampling of black-box systems. We focus on systems with high-dimensional, real-valued input variables and output patterns that can be transformed by some function into a scalar, real-valued quantity. Multi-objective problems can only be tackled by using a scalarization approach that combines many objectives into a single output function. We assume that we can efficiently generate input patterns to the black box. This implies that we know the specification of feasible inputs to the black box. We also assume that the black box can compute the output efficiently for all feasible input patterns, i.e., the black box returns a value within a realistic problem-dependent time span. We furthermore take for granted that the black box is oblivious to previously presented input patterns. This means that a current output of the black box only takes the current input into account and does not depend on the history of the input patterns. This, however, does not exclude the possibility of noisy output. We do not require that the black box always returns identical output for identical input. The output can be corrupted by (unknown) measurement or numerical noise or by any uncontrollable (unknown) input to the black box (for instance, human intervention). Important instances that fit this black-box definition are complex technical devices, computer algorithms, mathematical models, or scientific experiments. For such systems, simulation-optimization, (Bayesian) parameter identification or model reduction are common scientific tasks.

Throughout this thesis we conceptualize the black-box system as a *landscape*. Inspired by our shared visual experience of natural terrains and sceneries, we consider the real-valued input variables as a high-dimensional landscape domain. Neighborhood or nearness in this landscape domain must be provided by a suitable distance metric. We interpret the scalar output quantity as a height or elevation over the landscape domain. The landscape metaphor encourages a characterization of black-box systems in terms of topographical features such as valleys, ridges, mountain peaks, and plateaus. We introduce the notion of the *black-box landscape* in order to underline our view of black-box systems as high-dimensional, complex landscapes.

Many scientific disciplines use the landscape paradigm. The fitness landscape imagery is at the very heart of evolutionary biology and protein engineering. In molecular physics, the energy landscape perspective provides a unifying theme for understanding complex physical processes and phenomena. The landscape metaphor is also present in operations research, most prominently in the context of combinatorial optimization. All these fields influenced the present work in a number of aspects.

In the present work, we use *computation* as the fundamental scientific tool to examine black-box problems. The majority of the considered scientific questions is tackled by running computer simulations and inferring knowledge from the gathered empirical data. In a number of situations, we will, however, comment on known theoretical results or open mathematical problems.

## Main Contributions

We consider the following results as the main contributions of this thesis:

To the best of our knowledge, this thesis includes the first review of the landscape paradigm spanning the disciplines of biology, physics, chemistry, and optimization. Chapter 2 has been written because no adequate single reference could be found.

In the field of black-box optimization, we improve various aspects of the performance of the arguably best state-of-the-art optimizer, the Covariance Matrix Adaptation Evolution Strategy (CMA-ES). The general performance is increased by considering quasi-random instead of pseudo-random sampling. We introduce parallel CMA-ES schemes that can outperform standard CMA-ES for multi-funnel landscape topologies. We also revisit Gaussian Adaptation, a optimization and sampling scheme that has been largely neglected by the black-box optimization community. Our improved Gaussian Adaptation scheme shows remarkable performance on the considered benchmarks and ranks among the best known black-box optimizers.

An important conceptual result is that we can make an explicit link between black-box optimization and black-box (or indirect) sampling through Gaussian Adaptation. We show that the same idea of adaptation has emerged in these largely disparate fields, and we argue that a unifying framework for sampling and optimization might constitute an important contribution.

We also contribute to the fields of black-box characterization and black-box optimization benchmarking. We present a number of statistical landscape descriptors that can serve as features in a future statistical landscape classification framework. Novel benchmark problems are derived from geometry optimization of atomic clusters.

Finally, we revisit the configuration space of chain molecules with respect to a certain distance measure, the Root Mean Square Deviation (RMSD) after optimal superposition. RMSD is the most important distance metric in structural biology. We consider the simplest linear chain, the Random Walk model, that defines the base line for more complex polymer models. We quantify the neighborhood structure that is induced by the RMSD for the Random Walk model. Based on numerical results from CMA-ES black-box optimization runs, we are able to formulate a conjecture about an upper bound for the RMSD between any two Random Walks of arbitrary length.

## Previous Work

The present work is based on a number of previous scientific contributions. The characterization of continuous black-box landscapes has been influenced by Saltelli and co-workers in the field of sensitivity analysis (Saltelli et al., 2000), Lunacek and Whitley in evolutionary computation (Lunacek and Whitley, 2006; Lunacek et al., 2008), Stadler and co-workers in combinatorial optimization (Stadler, 1996; Reidys and Stadler, 2002), and Kauffman and Weinberger in theoretical biology (Kauffman and Weinberger, 1989; Weinberger, 1990). Our work in black-box optimization builds on two key sources: the works of Hansen (Hansen and Ostermeier, 1996; Hansen, 2000; Hansen and Ostermeier, 2001; Hansen et al., 2003) from the field of evolutionary computation, and Kjellström (Kjellström, 1969; Kjellström and Taxen, 1981; Kjellström, 1991; Kjellström and Taxen, 1992) from electrical engineering. Haario and co-workers (Haario et al., 1999, 2001), Neal (Neal, 2003), as well as Andrieu (Andrieu and Thoms, 2008) provide the foundation for our black-box sampling contribution. For cluster landscapes, an invaluable source of information is provided by the works of Wales (Wales, 2005). Cohn and Kumar's article introducing novel pair potentials has been instrumental for designing one of the presented benchmarks (Cohn and Kumar, 2009). The analysis of linear chain landscapes does not build on specific prior literature. It is a combined effort by Phillippe Hünenberger, Bojan Žagrović, and the author of this thesis (Müller et al., 2009).

## Structure of the Thesis

The remainder of this thesis is structured as follows:

### *Chapter 2: Landscapes*

Chapter 2 introduces the landscape paradigm. Starting from preliminary mathematical definitions we revisit the majority of landscape instances in physics, chemistry, biology, and combinatorial optimization. We also comment on black-box landscape properties and heuristic search. We note that such a review of the landscape paradigm in science does not exist in the scientific literature. Finally, the chapter introduces a list of model landscapes along with the IEEE CEC 2005 benchmark test suite. These are instrumental for the empirical performance evaluation of several methods.

### *Chapter 3: Characterization of Black-box landscapes*

In Chapter 3 we introduce a set of statistical black-box landscape descriptors that can probe different properties of a landscape, such as the global landscape topology, input separability, and landscape ruggedness. As a proof of concept we apply the descriptors to the CEC 2005 benchmark functions with known properties and analyze the quality of the estimators.

### *Chapter 4: Optimization of Black-box landscapes*

Chapter 4 presents the core of this thesis. We first present the state-of-the-art black-box optimizer CMA-ES and propose alternative restart and sampling schemes for it. We then introduce the concept of parallel CMA-ES and present the design and performance of one such instance, the Particle Swarm CMA-ES. Furthermore, we revisit the concept of Gaussian Adaptation and supplement the basic algorithm by a general-purpose parameterization, stopping criteria, and a restart strategy. All novel black-box algorithms are benchmarked on the benchmark test suite. Parts of this chapter are published in (Müller et al., 2009b; Müller and Sbalzarini, 2010c,b).

### *Chapter 5: Black-box Sampling*

Gaussian adaptation plays an important role in Chapter 5 as well. We show that minor changes in the algorithm turn it into to an adaptive Markov-Chain Monte Carlo sampler. We show the strengths and weaknesses of this novel black-box sampler on selected target distributions. Parts of this chapter are published in (Müller and Sbalzarini, 2010b; Müller, 2010).

### *Chapter 6: Atomic Cluster Landscapes for Black-box Optimization*

In Chapter 5 we consider geometry optimization of atomic clusters as novel benchmarks for black-box optimization. The proposed Cohn-Kumar and Lennard-Jones clusters exhibit different landscape topologies, thus spanning a wide range of problem difficulties. We argue that the presented problems should be included in future benchmark studies in order to improve the generality of black-box heuristics. Parts of this chapter are published in (Müller and Sbalzarini, 2009) or submitted for publication (Müller and Sbalzarini, 2010a).

### *Chapter 7: Analysis of Linear Chain Landscapes*

We consider linear chains in form of (self-avoiding) random walks in Chapter 7. We investigate the degree of inhomogeneity that is introduced in the random walk landscape domain by using RMSD as a distance measure. An extended version of this investigation is published in (Müller et al., 2009). We also investigate the maximum RMSD problem, which consists of finding the pair of structures that maximizes RMSD among all possible structures. Based on data from black-box optimization runs, we conjecture a closed-form upper bound for the RMSD between any two linear chains of the Random Walk type.

### *Chapter 8: Conclusion and Future Work*

We conclude this thesis in Chapter 8. We outline how the results of this thesis suggest several opportunities for future research, ranging from theoretical issues to concrete practical applications.

## Appendix

In the course of this thesis, we developed a number of well-tested software packages for black-box sampling and optimization. We present two software libraries in the Appendix: GaALib and pCMALib. GaALib comprises a set of MATLAB functions and scripts that implement all aspects of the Gaussian Adaptation scheme. It can be used for black-box optimization, sampling, and volume computation. pCMALib is a parallel FORTRAN90 library that implements both sequential and parallel CMA-ES in an efficient manner. All aspects of pCMALib are described in a manual-like style. We also present parallel scaling results that are published in (Müller et al., 2009a).

# 2
# Landscapes

> "Will you take us to Mount Splashmore?"
> Lisa and Bart Simpson, in: The Simpsons, Brush with Greatness, Episode no. 31, 1991

The notion of a landscape has been a valuable and highly influential concept in many areas of science. Inspired by our shared visual experience of natural terrains and sceneries, the landscape metaphor has been employed by researchers across disciplines to explain complex phenomena in a comprehensive manner. Sewall Wright introduced in his seminal paper (Wright, 1932) the concept of the adaptive or *fitness landscape* to modern evolutionary biology. Wright used fitness landscapes to illustrate the relationship between genetic or phenotypic traits of organisms and their associated evolutionary fitness. The evolution of a species can hence be imagined as an adaptive walk across hills and valleys of its fitness landscape, eventually settling around a peak. It is striking that more than 50 years later Stillinger and Weber employed a similar analogy to describe the packing structure and phase transitions in liquids and solids (Stillinger and Weber, 1984). Although the notion of the *potential energy surfaces* (PES) instead of *potential energy landscapes* (PEL) is used in the original article, Stillinger and Weber have the same metaphor in mind as Wright, this time, however, to explain the behavior of ensembles of atoms and molecules across thermodynamic regimes. Stillinger and Weber popularized the idea that static and (thermo-)dynamic features of molecular systems can be largely understood by analyzing the topography of the underlying energy landscape. A visual comparison of the original sketches from (Wright, 1932) and (Stillinger and Weber, 1984) emphasizes the strong similarity (Fig. 2.1).

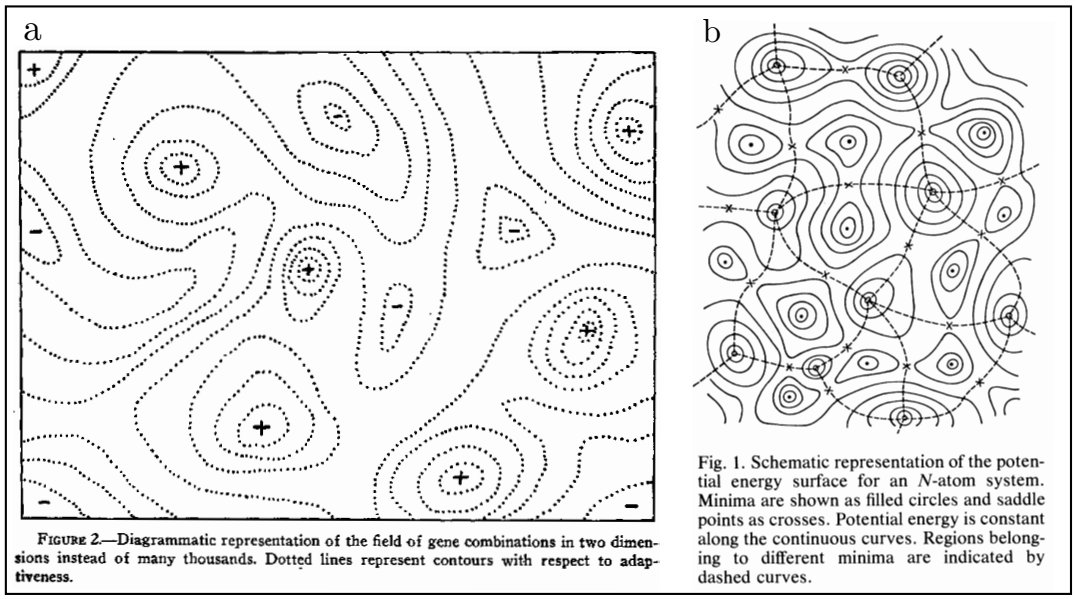



Figure 2.1: **a.** Sketch of a two-dimensional fitness landscape from Sewall Wright's 1932 publication on the role of mutation, inbreeding, crossbreeding and selection in evolution (Wright, 1932). **b.** Stillinger and Weber's sketch of a potential energy landscape of atomic systems (Stillinger and Weber, 1984).

Astonishingly few researchers, however, explicitly highlight the close conceptual relationship between fitness and energy landscapes. The works of Peter Stadler and co-workers are a notable exception (see, e.g., (Schuster and Stadler, 1994; Stadler and Stadler, 2002; Reidys and Stadler, 2002)) together with the influential review on ultrametricity in physics by Rammal and co-workers (Rammal et al., 1986) and Sherrington's introductory notes in a special issue on landscapes in Physica D (Sherrington, 1997). Despite the great value of metaphors and mental images in science, they can be, at the same time, subject of profound confusion due to their inherent subjective nature. As Peter Wolynes wrote in his philosophical article on landscapes (Wolynes, 2001): "Of all intellectuals, scientists are the most distrustful of metaphors and images. This, of course, is our tacit acknowledgment of the power of these mental constructs, which shape the questions we ask and the methods we use to answer these questions." It is, therefore, crucial that we first provide a formal definition of landscapes. We then present important landscape properties and geometrical concepts that allow a more refined view on landscape topographies and their practical use. After revisiting fundamental instances of the landscape paradigm in biology, (bio-)chemistry, and physics we eventually build a conceptual bridge to the field of optimization. Starting from a review of the landscape perspective in combinatorial optimization problems we develop a novel landscape perspective for black-box optimization. We close this chapter with an introduction of synthetic benchmark landscapes that are then used throughout this thesis.

## 2.1 Definitions and characteristics

We start with the most general definition of a landscape:

**Definition 2.1.** *A landscape $\mathcal{L}$ is a triple $(\mathcal{X}, d, f)$ consisting of*

1. *a set $\mathcal{X} \subseteq \mathbb{R}^n$,*
2. *a distance function d: $\mathcal{X} \times \mathcal{X} \to \mathbb{R}_0^+$,*
3. *a scalar function $f$: $\mathcal{X} \to \mathbb{R}$.*

Depending on the scientific context, the set (or domain) $\mathcal{X}$ has different composition and meaning. In evolutionary biology $\mathcal{X}$ can, e.g., be a finite set of genes. Each gene is a *sequence* or *string* of letters over the alphabet $\{A, C, G, T\}$ that represents the four different nucleotides. In a physical system, $\mathcal{X}$ may represent the positions of a collection of $n$ atoms in three-dimensional space. There, elements of $\mathcal{X}$ are often called *configurations*, *states*, or *micro-states* with $\mathcal{X} \subseteq \mathbb{R}^{3n}$. In optimization research, $\mathcal{X}$ represents the set of (feasible) solutions, e.g. binary strings or real-valued vectors of a certain dimension $n$. In this context, $\mathcal{X}$ is also termed *search space*. $\mathcal{X}$ can also represent the set of free parameters of a mathematical model, hence leading to the notion of a *parameter space*. In statistical models, $\mathcal{X}$ specifies *factors* or *input variables*.

The function $d$ adds structure to the domain $\mathcal{X}$. It can be, for instance, a mathematical *metric* (with the usual properties of non-negativity, positive definiteness, symmetry, and sub-additivity), a measure based on an order parameter in a physical system, or some other dissimilarity index. When $\mathcal{X}$ is, e.g., the set of binary strings of length $n$, a natural metric is the Hamming distance $d_{\mathrm{H}}$. The distance $d_{\mathrm{H}}$ is defined as the number of positions where the digits in two binary strings are different. The distance ranges between zero for identical strings and $n$ for strings that are different everywhere. The Euclidean distance $d_{\mathrm{E}}$ is often used when $\mathcal{X} \subseteq \mathbb{R}^{\mathrm{n}}$, where the range is again between zero for identical vectors and a maximum that is attained by the vectors defining the diameter $diam(\mathcal{X})$. For the three-dimensional unit cube $\mathcal{X} = [0, 1]^3$, the diameter in Euclidean distance is $\sqrt{3}$. It is noteworthy that the diameter of $\mathcal{X}$ is not always known *a priori* for a complicated domain or distance measure, thus hampering the interpretation of absolute distance values. This problem will be studied for linear chains in Chapter 7.

The function $d$ can also be defined by so-called *move sets*. A move set defines allowed moves or transitions from one element $\mathbf{x} \in \mathcal{X}$ to another $\mathbf{y} \in \mathcal{X}$ in a single step. A distance can then be defined by the minimal number of steps it takes to move from $\mathbf{x}$ to $\mathbf{y}$. In evolutionary biology, the move set could, e.g., be a point mutation in a gene per evolutionary time unit. In the binary string case the move set consisting of single bit flips is equivalent to the Hamming distance. We comment on the apparent connection between move sets and optimization algorithms in Section 2.3.1. In physics, the pair $(\mathcal{X}, d)$ is often called *configuration space* (Rammal et al., 1986; Reidys and Stadler, 2002), a term that plays a central role in Chapters 6 and 7 of this thesis.

The scalar function $f$, the third component of $\mathcal{L}$, is generally interpreted as height of the landscape. The height $f$ is a general mapping from the domain $\mathcal{X}$ to the real numbers. It assigns a real value to each element of $\mathcal{X}$. In evolutionary biology, $f$ is called the *fitness* and can, e.g., be experimentally measured by the reproductive success of an individual organism. In optimization, $f$ is called *fitness function*, *cost function*, or *objective function*. The standard term in physics is *energy*, most often *potential energy* or *free energy*, denoted by $h$, $E$, or $F$.

### *Black-box landscapes*

Definition 2.1 offers the most general definition of a landscape. In this thesis, we particularly focus on Black-box landscapes, defined as follows:

**Definition 2.2.** *A Black-box landscape $\mathcal{L}_B$ is the triple $(\mathcal{X}, d_X, f)$ consisting of*

1. *a set $\mathcal{X} \subseteq \mathbb{R}^n$,*
2. *a metric $d$: $\mathcal{X} \times \mathcal{X} \to \mathbb{R}_0^+$,*
3. *a scalar black-box function $f$: $\mathcal{X} \to \mathbb{R}$.*

In this thesis, we mostly consider landscapes whose support is a compact (convex) subset of the $n-$dimensional space of real numbers. Most often $\mathcal{X}$ is box-constrained, i.e. $\mathcal{X} = [\mathbf{l}, \mathbf{u}] \subset \mathbb{R}^n$ with the vectors $\mathbf{l}, \mathbf{u} \in \mathbb{R}^n$ defining the lower and upper bounds. In practice, even for unconstrained optimization problems box constraints are often imposed by the modeler in order to restrict analysis to certain $\mathbf{x}$ values of interest. These constraints also simplify certain mathematical operations, such as drawing uniform samples from the landscape domain.
A natural metric in such domains is the Euclidean distance $d_{\mathrm{E}}$:

$$d_{\mathrm{E}}(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})\ (\mathbf{x} - \mathbf{y})^T} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}\,, \tag{2.1}$$

with vectors $\mathbf{x}, \mathbf{y} \in \mathcal{X} \subseteq \mathbb{R}^n$, or the Mahalanobis distance

$$d_{\mathrm{M}}(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})\mathbf{C}^{-\mathbf{1}}(\mathbf{x} - \mathbf{y})^T}\,, \tag{2.2}$$

where $\mathbf{C}$ is a positive definite, symmetric matrix.
A central concept in this thesis is the notion of the *black-box function $f$*. The name implies that we consider $f$ a black box, i.e., the only information we can retrieve from $f$ is a real value for any given query $\mathbf{x} \in \mathcal{X}$. The black-box metaphor represents our general ignorance about the underlying system. Most often we do not know the analytic form of $f$, nor do we assume any mathematical property about $f$ such as, e.g., convexity or continuity. The function $f$ can be noisy, discontinuous, or non-differentiable. The concept of a black box conveniently addresses many problems of practical relevance in science and engineering. For example, researchers are often faced with fitting free parameters of a complicated mathematical or technical model in order to match the model output with noisy real-world measurements. The cost function

that measures the dissimilarity between model output and data can be considered a black-box function. Hence, the set of free parameters, a distance measure between parameter vectors, and the black-box function define a black-box landscape.

A similar yet more formal concept of a black box is known under the term *oracle* in theoretical computer science. Theoretical computer scientists "imagine an oracle as a device that solves certain problems for us, i.e. that, for any instance $\sigma$, supplies a solution $\tau$. We make no assumption on how a solution is found" (Grötschel et al., 1993) pp. 26.

One key problem with the landscape paradigm is our limited ability to comprehend and visualize more than three or four dimensions. The topography of geographical landscapes, such as the Swiss Alps, is completely specified by a two-dimensional coordinate system and a height or elevation associated with each point in the coordinate system. It comes as no surprise that the original fitness and energy landscape sketches in Fig. 2.1 are two-dimensional. Many landscapes in science are, however, high-dimensional. In order to fully appreciate the landscape metaphor even in the high-dimensional case, we have to resort to useful collective terms that are able to characterize landscape topologies and are, at least to some extent, measurable.

### *Landscape characteristics*

Landscape characteristics are such key topographic features that can be used to characterize high-dimensional landscapes. Although many of these features are not accessible in a black-box scenario we provide them here for completeness.

One property of landscapes is the *scale* with which the height or elevation $f$ varies over the whole domain $\mathcal{X}$. Do values of $f$ span over several orders of magnitude? Are they bounded from above or below? A comprehensive summary of the range of $f$ can be achieved by deriving or estimating the moments of the distribution of $f$ values, such as expectation values and variances, with respect to some measure. Under the assumptions that lower and upper bounds of the fitness range exist, these bounds correspond to the fitness values at the locations of the *global minimum* and *global maximum*, respectively.

**Definition 2.3.** *Let $\mathcal{X}$ be the domain of the landscape and the mapping $f : \mathcal{X} \to \mathbb{R}$. The mapping $f$ has a global minimum at $\mathbf{x}_{min}$ iff $f(\mathbf{x}_{min}) \leq f(\mathbf{x}) \forall \mathbf{x} \in \mathcal{X}$. The mapping $f$ has a global maximum at $\mathbf{x}_{max}$ iff $f(\mathbf{x}_{max}) \geq f(\mathbf{x}) \forall \mathbf{x} \in \mathcal{X}$.*

In general, global optima are hard to find. A considerable amount of work in this thesis is dedicated to the efficient search for global minima of black-box landscapes, as we will see in Chapter 4. Even in cases where the locations of global optima are known, they often provide only limited information about the overall geometric topology of the landscape surface. In many cases, it is easier and more informative to analyze features associated with *local optima*.

**Definition 2.4.** *Let $\mathcal{X}$ be the domain of the landscape and the mapping $f : \mathcal{X} \to \mathbb{R}$. Let $\mathcal{N}(\mathbf{x})$ be the neighborhood of $\mathbf{x}$. Then $f$ has a local minimum at $\mathbf{x}^{loc}$ iff $f(\mathbf{x}^{loc}) < f(\mathbf{x}) \forall \mathbf{x} \in \mathcal{N}(\mathbf{x}^{loc})$. The mapping $f$ has a local maximum at $\mathbf{x}^{loc}$ iff $f(\mathbf{x}^{loc}) > f(\mathbf{x}) \forall \mathbf{x} \in \mathcal{N}(\mathbf{x}^{loc})$.*

The neighborhood $\mathcal{N}(\mathbf{x})$ is induced by the distance measure $d$. For example, when $\mathcal{X}$ is the set of binary strings of length $n$ and $d_{\mathrm{H}}$ the Hamming distance then we can define $\mathcal{N}(\mathbf{x}^{\mathrm{loc}}) = \{\mathbf{x} \in \{0,1\}^n \mid d_{\mathrm{H}}(\mathbf{x}, \mathbf{x}^{\mathrm{loc}}) = c\}$. In the simplest case $c = 1$, which we call the 1-neighborhood. The distance, and hence the neighborhood, can also be defined via more complicated move sets. When $\mathcal{X} \subseteq \mathbb{R}^n$ and $f$ is a smooth function with continuous first and second derivatives, a local minimum/maximum can be characterized by the usual optimality conditions:

**Definition 2.5.** *Let $\mathcal{X} \subseteq \mathbb{R}^n$ and $f : \mathcal{X} \to \mathbb{R}$. Then $f$ has a* stationary point *at $\mathbf{x}^{loc}$ iff $\nabla f(\mathbf{x}^{loc}) = \mathbf{0}$ where $\nabla f$ denotes the n-dimensional gradient vector with components:*

$$\nabla f(x_i^{loc}) = \frac{\partial f}{\partial x_i}\,. \tag{2.3}$$

*The mapping $f$ has a local minimum at $\mathbf{x}^{loc}$ if, in addition, the Hessian $n \times n$ matrix $H(\mathbf{x}^{loc})$, the symmetric matrix of second derivatives with elements*

$$H_{i,j}(\mathbf{x}^{loc}) = \frac{\partial^2 f(\mathbf{x}^{loc})}{\partial x_i \partial x_j}\,, \quad i,j = 1,\ldots,n \tag{2.4}$$

*is positive definite. For a local maximum $H$ must be negative definite at $\mathbf{x}^{loc}$. A first-order saddle point is located at $\mathbf{x}^{loc}$ if $H$ has n-1 positive eigenvalues and exactly one negative eigenvalue.*

In the context of optimization problems or probability distributions the optima are often called *modes* leading to the term *multi-modal landscapes* in the presence of multiple optima. In the following we focus on using a collection of local minima for landscape characterization. The argumentation also holds for local maxima for a landscape with negated $f$. Consider a landscape with multiple local minima and imagine that we have an algorithm that finds for each point in the domain a local minimum by following the steepest descent path. Then we can tesselate the domain of the landscape into disjoint regions, each containing the set of points that lead to the same local minimum. We refer to such a region as *catchment basin*, *basin of attraction*, or just *basin*. Figure 2.2 shows a sketch of a landscape tessellated into 6 basins. In the following we give a formal definition:

**Definition 2.6.** *Let $\mathcal{L} = (\mathcal{X}, d, f)$ be a landscape with $K$ local minima $\mathbf{x}^{(k),loc}$ with $k = 1,\ldots,K$. Let $\mathcal{M}$ be an algorithm that proceeds from each $\mathbf{x} \in \mathcal{X}$ to a local minimum via the steepest descent path. Then the catchment basin $\mathcal{C}_k$ is defined as:*

$$\mathcal{C}_k = \{\mathbf{x} \in \mathcal{X} \mid \mathbf{x}^{(k),loc} = \mathcal{M}(\mathbf{x})\}\,. \tag{2.5}$$

*The boundary $\mathcal{B}_k$ of basin $\mathcal{C}_k$ is the set of all $\mathbf{x}$ that have at least one point $\mathbf{y} \in \mathcal{N}(\mathbf{x})$ with $\mathbf{y} \in \mathcal{C}_i, i \neq k$.*

One situation hampers the generality of the basin definition on smooth landscapes: the existence of regions of constant $f$ or *landscape neutrality*. In such regions a steepest descent algorithm fails to proceed. We assume that the algorithm $\mathcal{M}$ has a mechanism to detect these regions. The algorithm then assigns points on the plateau to an arbitrary neighboring basin. On black-box landscapes the situation is even more complicated when $f$ is noisy or

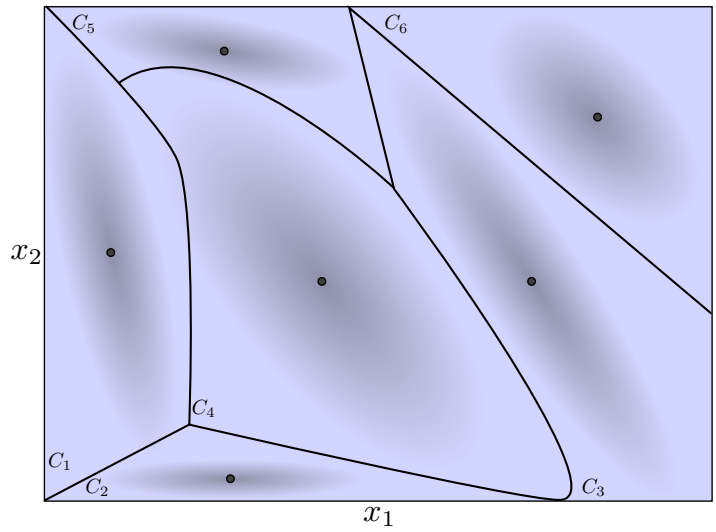



Figure 2.2: Sketch of a tesselation of a 2D domain into 6 basins of attraction $\mathcal{C}_k$. The black dots (•) represent the corresponding minima, the lines the boundaries separating the basins.

discontinuous. We also assume that the (black-box) algorithm $\mathcal{M}$ can handle these instances and that $\mathcal{M}$ provides a unique basin assignment.

Two important properties of a basin are its depth and its size or volume. We define *basin depth*, sometimes also termed *barrier height*, in the following way:

**Definition 2.7.** *Let $\mathcal{L} = (\mathcal{X}, d, f)$ be a landscape with $K$ local minima $\mathbf{x}^{(k),loc}$, catchment basins $\mathcal{C}_k$, and boundaries $\mathcal{B}_k$ with $k = 1, \ldots, K$. The basin depth, or barrier height, $T(\mathcal{C}_k)$ is defined as*

$$T(\mathcal{C}_k) = \min_{\mathbf{x} \in \mathcal{B}_k} f(\mathbf{x}) - f(\mathbf{x}^{(k),loc}) \quad k = 1, \ldots, K\,. \tag{2.6}$$

When $\mathcal{X} \subseteq \mathbb{R}^n$ and $f$ is a smooth function with continuous first and second derivatives, $T(\mathcal{C}_k)$ is the fitness difference between the basin minimum and the lowest-lying saddle point or maximum separating the minimum from a neighboring basin. Although the notion of basin size or volume is intuitive, a clear mathematical definition depends on the properties of $\mathcal{X}$. Size could, e.g., be defined as the number of configurations within a basin when $\mathcal{X}$ is a finite set.
Partitioning the landscape domain into basins of attraction offers many insights into landscape topology. A complete tessellation would enable use to derive the total number of minima of the landscape, including global minima. Another property is the distribution of basin depths and sizes. Interesting questions related to basin depths and sizes are: What is the distribution of basin depths compared to the global range of $f$ values? Does the landscape domain contain many basins of the same size or are there dominating large basins and a few small ones? The specific distribution of basin sizes could, e.g., tell us how likely it is to hit the basin that includes the global minimum. The number of basins and their size distribution are

also related to the effort for an algorithm to enumerate all basins. More complex landscape features account for the spatial arrangement of the basins and its relation to (i) the height of the associated local minima and (ii) the basin depths. A prominent example is the hierarchical arrangement of basins into *super-basins* or *funnels*. A one-dimensional sketch of a funneled landscape is shown in Fig. 2.3. A funnel is characterized by the following properties: (i)

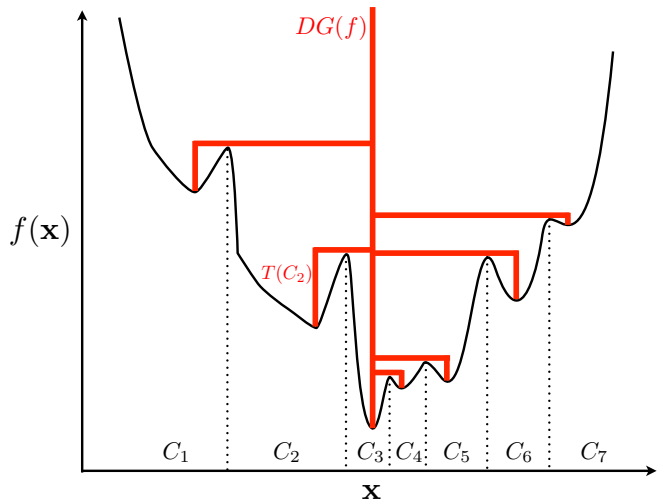



Figure 2.3: Sketch of a funneled landscape in 1D. The dotted lines mark the transition regions between the basins $\mathcal{C}_k$. $\mathcal{C}_3$ contains the global minimum and is the funnel "bottom". The height of the minima in neighboring basins decreases toward the funnel bottom. The corresponding disconnectivity graph (DG) is shown in red. As an example the minimal basin depth is labeled for basin $\mathcal{C}_2$.

The depth of individual basins within a funnel is considerably smaller than the total range of $f$ across the whole landscape. (ii) The height of the minima of the basins decreases with decreasing distance to the center or bottom of the funnel. If **all** basins in a landscape are arranged in this way, we call it a *single-funnel landscape*, otherwise a *multi-funnel landscape*. When knowledge about basin arrangement, local minima height, and basin depth is known, the landscape can be visualized using *disconnectivity graphs* (DG) or *barrier trees* as shown in Fig. 2.3. The vertical axis corresponds to the fitness or energy scale, the horizontal axis to a general coordinate that is able to separate the different minima. Lines are drawn upward starting from every local minimum. The lines of neighboring basin are joined to the trunk of the tree (or an internal node) at the $f$ level that corresponds to the height of the basin minimum plus the basin depth. For physical systems, the shape of the resulting tree can offer insights into thermodynamic properties, as we will see in Chapter 6.

We emphasize again that the notion of a funnel is rather a metaphor than a precise mathematical object. We do not specify how exactly the global range of $f$ values and the basin depths have to be related, nor do we prescribe how distance to the funnel bottom and the decrease of minima height are correlated. We will, however, encounter numerous examples

of funneled landscapes where we give more details about specific funnel structures and their implications for landscape characterization and optimization.

A complete partitioning of the landscape domain into basins along with knowledge of local minima height, basin depth, and basin volume would provide an almost complete characterization of the landscape. This ideal situation, however, is almost never achievable in practice. The number of minima (and hence the number of basins) of many landscapes scales exponentially in the problem dimension. Hence, even enumerating all minima cannot be achieved in polynomial time or space. Researchers have to either restrict the landscape partitioning to a small subregion of the domain or have to resort to techniques that describe landscapes in a coarser way. A ubiquitous notion is *landscape ruggedness* or *landscape roughness*. Ruggedness is an intrinsically local property of a landscape that can vary across the domain. In the general case, we imagine the degree of ruggedness of a subdomain of a landscape as the presence or absence of correlation between the fitness values of neighboring points in the subdomain. For some landscapes the correlation length can be determined analytically, for others it has to be estimated empirically. Note that the distance defined on the landscape plays a decisive role for ruggedness. It is conceivable that the landscape is considerably rougher for some distance measure than for others. If knowledge about the locations of local minima is at hand, ruggedness can also be related to the number of minima within a landscape subdomain.
Ruggedness is an average property over a certain neighborhood and hence does not account for how individual coordinates or certain combinations of coordinates influence $f$. For example, it might be valuable to determine how a change in individual variables is related to a change in fitness. This is the scope of *sensitivity analysis*. Sensitivity analysis studies how the variation in landscape height can be apportioned, qualitatively or quantitatively, to different sources of variation in the landscape domain. Both local and global techniques are available. If a landscape represents some parameterized mathematical model, then sensitivity analysis can also be seen as a way to provide information about the importance of model parameters. Several sensitivity analysis techniques also allow quantifying the interaction structure between variables. For some landscapes, changing one variable will not affect the effect of other variables on the fitness. Such landscapes are called *separable*. Consider an $n$-dimensional landscape domain. The separability property then allows characterizing each dimension independently and then combining the $n$ one-dimensional characteristics into a global $n-$dimensional one. Searching for the global minimum in an $n$-dimensional separable landscape amounts to solving $n$ one-dimensional minimization problems. Note that for discrete landscapes non-separability is also related to *epistasis*, an important concept in genetics. Epistasis is the phenomenon where the effects of one gene are modulated by one or several other genes.

## 2.2 Landscape paradigms in science

The landscape paradigm has been so influential in modern science that it is instructive to briefly present the most important landscape instances. We first outline landscapes in biology covering a wide range of length and time scales.This leads to a natural transition to the energy landscapes in physics and chemistry that are subject of Section 2.2.2. This section also serves to link two key aspects of this thesis: landscapes and optimization algorithms.

### 2.2.1 Landscapes in biology

The landscape metaphor has been introduced by Sewall Wright in his presentation at the Sixth International Congress of Genetics in 1932. Wright was an evolutionary biologist and created the adaptive landscape picture to illustrate his "Shifting Balance Theory" of evolutionary change. The details of this controversial theory are beyond the scope of this thesis and can be found in (Wright, 1932). Important to us, however, is the way Wright imagined the phenomenon of evolution as a dynamic process over a landscape. His key idea was the following: Imagine a population of a species, each having a collection of different genes shaping the "genotypic space" $\mathcal{X}$. Each individual in the population comprises an instance of these genes at a specific point in time. This list of genes, the genotype, is a point $\mathbf{x}$ in the high-dimensional space $\mathcal{X}$. Each gene collection induces a specific phenotype that is associated with a certain evolutionary fitness $f(\mathbf{x})$. Hence, the population of individuals can be imagined as a point cloud on a fitness surface over the genotypic space. The population can move in the genotypic space by sequential genetic changes (*mutation* or *recombination*) over several generations. When *natural selection* acts on the population over time, only individuals with high fitness may survive. Ultimately, the population will cluster (or "adapt") around fitness peaks in the landscape. In Fig. 2.1a these peaks are denoted by a +. A closer look at this figure also reveals that Wright did not label the axes of the landscape domain. Wright was aware that there is no easy way to project the high-dimensional genotypic space into two or three dimensions. For him, the landscape was just a metaphor. He imagined that individuals with similar genotypes should be close in this landscape and have, at least to a certain extent, similar fitness. Unfortunately, neglecting the axis labels in his original landscape sketch caused considerable confusion throughout the scientific community until today. Kaplan even advocates in his recent philosophical paper "The end of the adaptive landscape metaphor?" (Kaplan, 2008) that the fitness landscape picture should be given up entirely and replaced by more formal modeling, even at the expense of being less intuitive. In our view, this criticism is only partially valid as there have been many attempts to formalize the landscape concept more rigorously. Gillespie introduced the *mutational landscape* in this article "Molecular Evolution over the Mutational Landscape" (Gillespie, 1984) which challenges Wright's evolutionary theory while keeping the landscape metaphor. The mutational landscape is an alternative model for molecular evolution based on extreme value theory that, with some modifications, is highly valuable to explain data from real molecular evolution experiments, such as single-stranded virus DNA (Rokyta et al., 2005).

The landscape paradigm has not only been useful in evolutionary biology. It conquered branches of biology that investigate systems on totally different time and length scales. The most prominent example arose in the context of "epigenetics" in developmental biology. The word "epigenetics", coined by Conrad Hal Waddington, was used to describe events that could not be understood by genetic principles. Waddington defined epigenetics as "the branch of biology which studies the causal interactions between genes and their products, which bring the phenotype into being" (Waddington, 1942). In 1957, he proposed the metaphor of an *epigenetic landscape* to illustrate the process of cellular decision-making during development. Waddington's original sketch is shown in Fig. 2.4. Waddington used this picture to illustrate how cells, despite their identical genetical material, can nonetheless differentiate into different

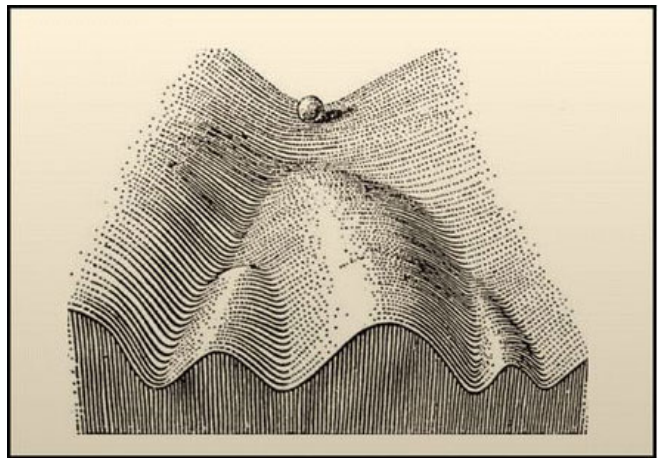


Figure 2.4: Waddington's original sketch of the epigenetic landscape (Waddington, 1957). The marble represents a cell in early developmental stage. The specific trajectory it takes on the landscape will irreversibly lead to a local minimum in the landscape that determines its "fate", i.e., into what tissue type it will differentiate.

phenotypes due to epigenetic modifications. The collection of possible modifications acting on cells form an epigenetic landscape with valleys and ridges. A cell during developmental phase can take specific permitted trajectories, leading to local landscape minima that define different phenotypes or cell fates. This metaphor is very lively even today. A modern view of the molecular mechanisms underlying the epigenetic landscape can be found in Goldberg and co-workers' recent essay "Epigenetics: A Landscape takes Shape" (Goldberg et al., 2007).

One of the most fascinating biological systems where the landscape metaphor had a lasting impact are *proteins*. Proteins are ubiquitous in living organisms and are considered the building blocks of life. They exhibit an amazing variety of three-dimensional structure, size, and biological function. Proteins are *chain molecules* composed from 20 distinct *amino acids*, each having different biophysical properties. The specific sequence of the amino acids in the chain dictates the three-dimensional structure or shape of the protein. Key structural motifs in proteins are helices, sheets, and coils (see Fig. 2.5 for an example). The "central dogma" of molecular biology (Crick, 1970) states that proteins are the sole products of the inherited genetic information. Each gene, a string of some length over the alphabet $\{A, C, G, T\}$, codes for a specific protein. A gene is partitioned into triplets, called *codons*. Each codon is specific for a distinct amino acid. Because only 20 different amino acids occur in real proteins out of the $4^3 = 64$ possible codons, the genetic code is said to be degenerate. By the time Wright introduced the fitness landscape paradigm, all this detailed molecular information was not known. It was not until 1970 when John Maynard Smith took up Wright's evocative landscape imagery and created the concept of a "protein space" (Maynard-Smith, 1970). Back then, there was considerable debate about the tremendous disparity between the number of naturally occurring proteins and the much larger number of genetically encodable proteins.

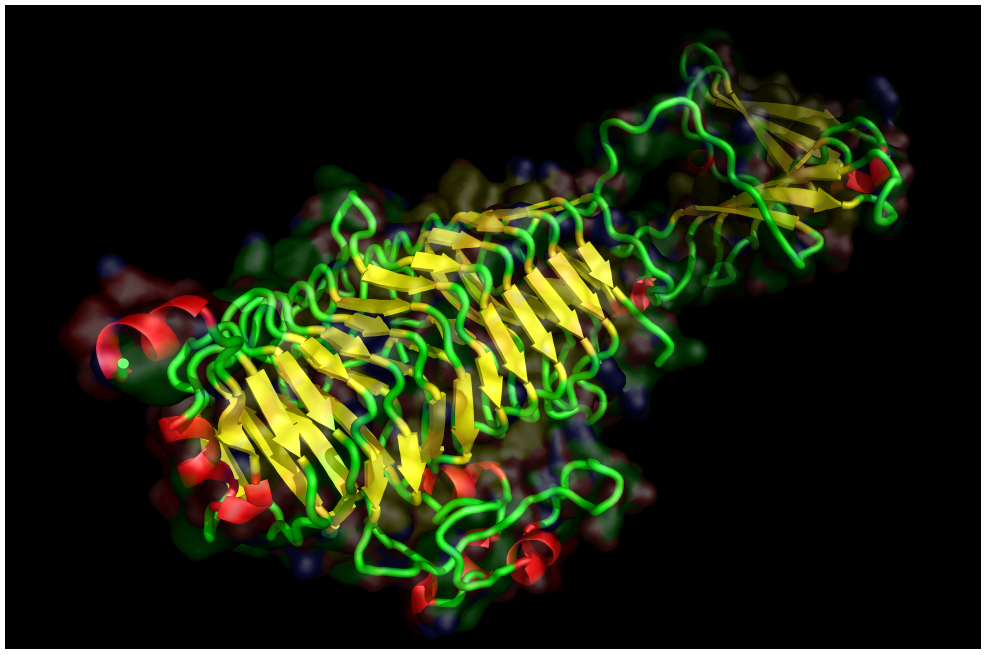


Figure 2.5: Cartoon representation of the Phage 22 tail spike protein (PDB entry 1TYU), consisting of helices (red), sheets (yellow), and coils (green) as structural elements.

Maynard Smith's seminal idea was to envision protein evolution as an adaptive walk through protein space, where functional proteins may change to fitter variants through single amino-acid substitutions. Fitness can be any specific functional property, such as the capacity to catalyze a specific reaction or bind a specific ligand. For a protein of length $n$ the space comprises $20^n$ elements. If neighborhood is defined by site-specific single amino-acid differences, each protein has $19n$ neighbors. Due to the fact that mutations occur on the nucleotides in the genes, the degeneracy of the genetic code implies that the protein fitness landscape is intrinsically different from its genetic counterpart, most prominently through the existence of large regions of fitness neutrality. This observation led the japanese biologist Motoo Kimura to the formulation of his "neutral theory of molecular evolution" in the late 1960's, a corner stone of modern evolutionary biology (Kimura, 1983).

The concept of a protein fitness landscape is, however, more than just a useful metaphor. Over the past decades this imagery catalyzed an incredible amount of both theoretical and experimental research. Due to the fact that many biological and chemical properties of proteins are accessible in laboratory experiments, the possibility emerged to test and validate theoretical landscape models through controlled measurements. One of the earliest attempts was made by Kauffman (Kauffman, 1993) and further developed by Kauffman and coworkers in the context of the immune response of higher organisms (Kauffman and Perelson, 1988; Kauffman and Weinberger, 1989). One fundamental task within the complex immune system machinery is to efficiently identify molecules, so-called antigens, that have not been produced

natively by the body. To this end, the organism has to evolve specific proteins (antibodies) that have a high binding affinity to intruding antigens. This process is called maturation. Kauffman and co-workers used the *NK-model* of a rugged fitness landscape for the maturation of the immune response. Inspired by physical spin glasses, Kauffman and Weinberger defined the model in the following way: "The NK model is meant to apply to systems of many, N, parts, where the functional contribution of each part depends upon the "state", among A alternatives, of that part, and is epistatically affected by an average of K other parts" (Kauffman and Weinberger, 1989). More formally, the NK-model is intended to capture interactions between the bits in a binary string (alleles, chromosomes, proteins), giving rise to landscapes that are tunable in terms of epistasis and ruggedness. This model can be used to describe the epistatic nature of genes. In the context of the immune response, they chose the system to be the variable (V) region in an antibody molecule. The N parts are the amino acids with A=20 alternatives per state and the fitness of a V region is its binding affinity for an incoming antigen. They imagined the maturation of the immune response as an adaptive walk on an "affinity landscape". The NK-model provided a means of abstracting the complex immune response process. It predicted, through careful design of a particular model instance and parameter tuning, a number of qualitative features of antibody affinity evolution, such as the speed of adaptation and the existence of conserved patterns within the V region (Kauffman and Weinberger, 1989). For a recent study where the NK model has been successfully used for the analysis of a DNA-protein affinity landscape, we refer to (Rowe et al., 2010). We comment on more formal optimization studies on the NK model in Section 2.3.

Binding affinity is not the only property that can determine the evolutionary fitness of a protein and, hence, its associated landscape. For enzymatic proteins, fitness might be defined as the capacity to catalyze a specific reaction. Maintaining structural and functional stability over a wide range of temperatures might also be a valuable objective. The evolution of antibiotic resistance in bacteria through specific enzymes is another example (Carneiro and Hartl, 2010). For decades, molecular biologist have sought after tools and techniques to decipher the complex interactions between the composition and organization of proteins and any of its functional properties. Despite paramount advances in some areas of protein research, a true molecular-level understanding of why one protein performs a certain task better than another remains largely elusive. This fact also hampers the possibility of rationally engineering and improving protein functions for dedicated biological or pharmaceutical purposes. It is amazing that the idea of evolutionary optimization, which has been so successful in technical applications, has been brought back to biology through the method of *directed evolution*. In their excellent review "Exploring protein fitness landscapes by directed evolution", Romero and Arnold state: "Directed evolution circumvents our profound ignorance of how a protein's sequence encodes its function by using iterative rounds of random mutation and artificial selection to discover new and useful proteins. Proteins can be tuned to adapt to new functions or environments by simple adaptive walks involving small numbers of mutations." (Romero and Arnold, 2009). Directed evolution is, hence, an experimental tool for finding local maxima in the protein fitness landscape. A typical iteration of a directed evolution experiment involves three steps: diversification, selection, and amplification. The first step is concerned with the creation of a diverse pool of candidate proteins by randomization techniques such as DNA shuffling or error-prone PCR. The second step uses screening techniques to isolate and

select candidates with improved functional properties. Finally, the identified candidates are subject to a replication process that increases their abundance by orders of magnitude. The abundant protein candidates are then subject to biochemical analysis and re-entry into the directed evolution cycle in the next round. The directed evolution technique established itself as a standard tool in molecular biology labs since the mid-nineties. Interestingly, Rammal and co-workers already commented on the possibility of such a technique under the term "evolutive biotechnology" or "simulated evolution" (Rammal et al., 1986), where they refer to presentations by P.W. Anderson and Manfred Eigen at the 1984 Founding Workshops of the Santa Fe Institute (Pines, 1988). Rammal et al. also point out that for proteins the landscape picture naturally arises not only on the evolutionary but also on the molecular scale: The structure and dynamics of protein chains are governed by atomic and molecular forces within the chain and its interaction with the environment. It has been proven extremely fruitful to picture the dynamical behavior of a protein as a walk across a complex high-dimensional *energy landscape*, a perspective we consider next.

### 2.2.2 Energy landscapes in chemistry and physics

The concept of an *energy landscape* is fundamental to many areas of (bio-)chemistry and physics. Imagining the dynamical behavior of a molecular system as a process over a high-dimensional energy landscape has been a key concept to elucidate complex patterns occurring in nature. In protein biophysics, the energy landscape perspective is intimately connected with one of the grand challenges in the area: *the protein folding problem*. What is this grand challenge? We have introduced proteins as concatenations of amino acids, forming linear chains. In a living cell, proteins are produced by ribosomes. When a protein leaves this molecular machinery, its configuration is more or less unstructured and it is considered to be in the *unfolded* or *denatured state*. A protein's function, however, is largely determined by its three-dimensional shape. In order to attain this functional conformation, a protein undergoes considerable configurational rearrangement, a process called protein folding (see Fig. 2.6). Since Christian B. Anfinsen's Nobel prize-winning experiments, it is known that, at least for small globular proteins, the *native* or *folded structure* is determined only by the protein's amino acid sequence (Anfinsen, 1973). Anfinsen's dogma, sometimes also called the thermodynamic hypothesis, states that at the environmental conditions at which folding occurs, the native structure is a stable, unique, and accessible minimum of the energy surface. For many years, it has been a mystery how a protein with its huge number of conformational degrees of freedom can find this stable minimum at high speeds that are observed in laboratory experiments. Cyrus Levinthal has formulated this conundrum in his famous thought experiment, Levinthal's paradox. Imagine a protein chain consisting of 100 amino acids. Each bond between amino acids, the peptide bond, defines two angles between consecutive amino acids. In Levinthal's model, each of these angles can attain only three possible values. Even this simplified chain has already $3^{99}$ possible configurations, more than the number of atoms in the universe. It seems thus rather unlikely that a protein finds the configuration corresponding to the stable minimum by random sampling. Also, in real proteins, the degrees of freedom are continuous variables, and the previous discretization does not hold. The "paradox" that most small proteins fold spontaneously in milliseconds or even microseconds, despite the huge conformational space, arose from the imagination that the energy landscape guiding the search to

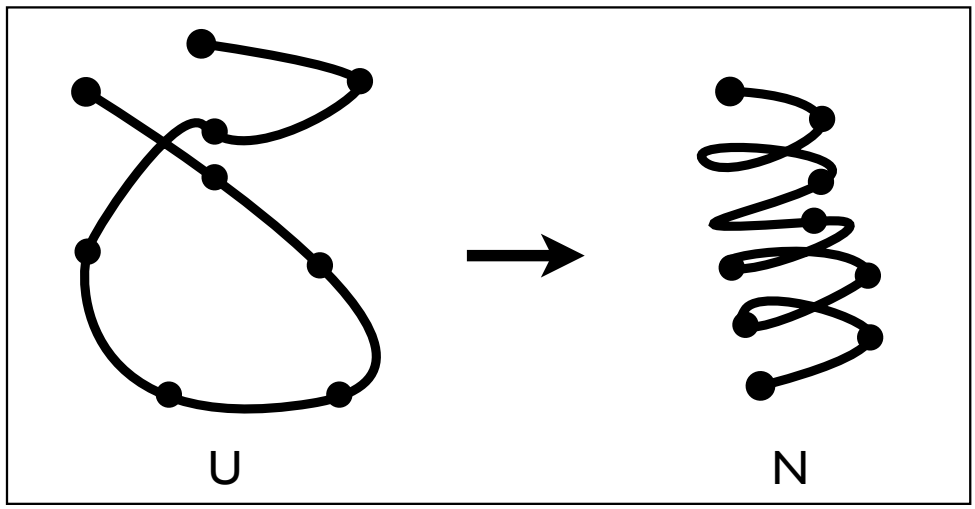



Figure 2.6: Sketch of the protein folding process. A long unstructured chain, the unfolded state (U), transforms into a folded native (N) state, e.g., a helical structure.

the stable configuration looks like a "golf course", as depicted in Fig. 2.7 a. Finding the native conformation seemed much like finding the proverbial "needle in the haystack". Levinthal's paradox has been resolved by the intriguing hypothesis of a "funneled energy landscape" (see Fig. 2.7 b). Starting from theoretical works by Hans Frauenfelder, Joseph Bryngelson, and Peter Wolynes in the 1980's (Bryngelson and Wolynes, 1989; Frauenfelder, H. and Sligar, S. G. and Wolynes, P. G., 1991), this hypothesis culminated in what some researchers called the "new view" on protein folding (Dill and Chan, 1997). The old view on protein folding considered the metaphor of "pathways" that proteins take from the unfolded to the native state. The new view was associated with the funneled energy landscape perspective that explained the folding process as the guided movement of denatured conformations starting from the "rim" of a funnel down to the bottom where the native state was located. Both the review by Dill (Dill and Chan, 1997) and the essay by Wolynes (Wolynes, 2001) excellently summarize the implications of the energy landscape perspective for protein folding research. The strength of the funneled-landscape hypothesis is that it provides guidance for the implementation and interpretation of laboratory protein folding experiments. Features of the folding funnel can nowadays be probed experimentally (Mello and Barrick, 2004). The speed of folding can be measured and associated with the ruggedness or roughness of the energy landscape (Nevo et al., 2005; Kapon et al., 2008). The importance of a funneled energy landscape has also been recognized for the assembly of multi-domain proteins, i.e., proteins that consist of different modular regions (Faraldo-Gomez and Roux, 2007). Recent studies extend the energy landscape perspective from single-protein folding to protein-protein interactions (Hunjan et al., 2008). Moreover, Clark advocates that the single folding-funnel perspective needs to be extended to a double-funnel topology in order to fully account for the behavior of proteins under physiological conditions (Clark, 2004). The second funnel represents protein aggregates, i.e.,

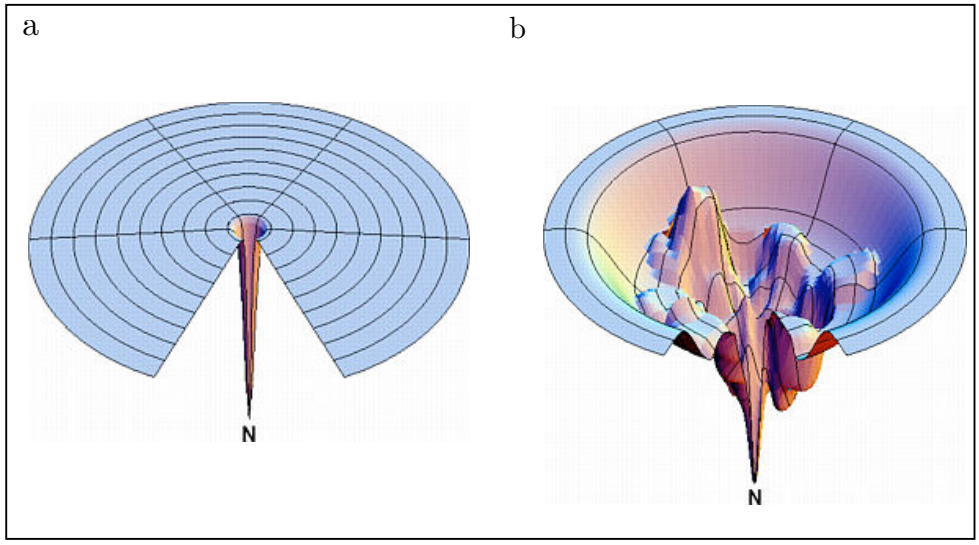



Figure 2.7: **a.** Sketch of Levinthal's "golf-course" landscape. **N** represents the region of the native state of the protein that is surrounded by a flat energy plateau. **b.** A rugged but funneled energy landscape that surrounds the native state (N). Both figures/Landscapes are taken from (Dill and Chan, 1997).

clumps of dense protein configurations that are neither in the native state nor in the unfolded state but are often observed in experiments (see Fig. 2.8).

Besides protein folding and the physical perspective on protein dynamics, there are also other branches of chemistry and physics that have adopted the landscape metaphor, most prominently in studies of liquids and solids. We have already referred to the paper by Stillinger and Weber (Stillinger and Weber, 1984) that contained probably the first sketch of a multi-dimensional energy surface (Fig. 2.1b). However, some ideas presented in that paper can be traced back at least to the 1960's (see references in (Stillinger and Weber, 1984) and (Goldstein, 1969)). What is the purpose of the energy landscape perspective in liquid or solid-state systems, and what are the differences to the previous protein energy landscape picture? In our considerations of protein folding we have avoided to clearly define which physical energy we actually mean. Because folding takes place at physiological temperature, the native state is a minimum of a *free energy* surface. The free energy is a thermodynamic concept that includes two contributions, the *potential energy* or *Hamiltonian* of the system and the *entropy*. It is beyond the scope of this thesis to give an introduction to thermodynamics, but we rather convey the general idea. In essence, the potential energy includes all energetic terms arising from the interactions between atoms in a given state, such as the Coulomb and Van der Waals energies. Entropy measures the flexibility of the system to adopt different molecular configurations at finite temperature $T > 0$. It is hence a property of an *ensemble* of configurations rather than a single one. At absolute zero temperature ($T = 0$) the free and the potential energies coincide. Because proteins, which can be considered a particular

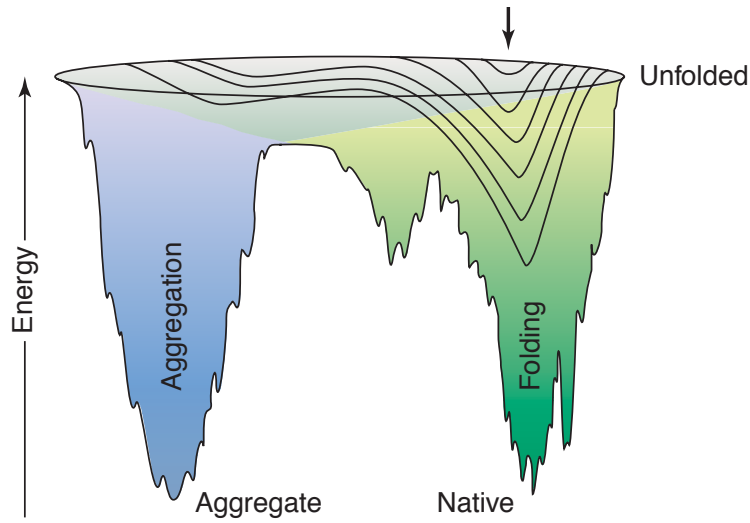



Figure 2.8: Clark's double-funnel perspective on protein folding in the cellular context. The first funnel (green) shows the traditional folding funnel. The second funnel (blue) represents non-native protein aggregates that are often observed in experiments (Clark, 2004).

instance of soft matter, operate at physiological temperature ranges, entropic contributions cannot be neglected. The protein folding process must hence be considered as a minimization over the free energy surface. For simpler forms of solid matter, such as pure substances in bulk or small atomic clusters, Stillinger and Weber advocated the potential energy landscape as a unifying concept for a deeper understanding of atomic arrangements. From a wide variety of experimental techniques it had been known that periodic crystalline order provided the most stable arrangement for many pure substances in the solid phase. While perfect crystals are a rarity in our natural environment, many kinds of dense matter appear as more or less defective crystals, i.e. arrangements that show regular packing with voids or interstitials at some lattice positions. Stillinger and Weber's ingenious idea was to envision these arrangements as local minima of the potential energy landscape. Consider a system of $N$ atoms in three dimensions at $T = 0$. The potential energy $E$ is hence a function of $3N$ atom positional coordinates. Depending on the physical properties of the atoms the system may exhibit different stable packings, the so-called *inherent structures* of the system (Stillinger and Weber, 1984). The height of the potential energy barriers between inherent structures, the number of transition paths over saddles between them, and the overall topography of the PEL can be used to explain the melting and freezing behavior of the system at finite temperature. The landscape perspective also offers a way why some materials are "structure seekers" and others are "glass formers". When cooled down at a certain rate, some materials form regular packing structures, whereas others relax to a disordered state that lacks periodicity but behaves like a solid. The reason why this is so has been a long-standing riddle in physical chemistry. From an energy landscape perspective such behavior is conceivable, and R. Steven Berry and co-workers introduced the notion of structure seekers and glass formers (see e.g. citeBall:1996), which made

the semantic meaning precise in terms of the energy landscape (Cox et al., 2006): "In short, "structure-seeker" means "able to relax to one of a set of structures very small compared with the set of all local minima", and "glass-former" means "relaxes to any of a very large fraction of the available local minima"." The global topology of the energy landscape of a structure seeker can be imagined as a single funnel, whereas glass formers have a multi-funnel or unfunneled topology. Numerous experimental and computational studies have adopted the energy landscape perspective for a wide variety of molecular systems. Numerical studies nowadays use both classical and quantum-mechanical energy formulations for different bulk materials and clusters. A famous instance are Lennard-Jones clusters, i.e., clusters of up to 200 atoms that interact via the Lennard-Jones (LJ) pair potential. This pair potential is the simplest model for the interactions between noble-gas atoms such as Argon. We will consider the landscapes arising from this potential in Chapter 6. Recent studies also applied inherent structure analysis to simplified protein models (Kim and Keyes, 2007) and all-atom models of proteins (Rao and Karplus, 2010), hence closing the circle of the two landscape paradigms presented here. A superb summary of energy landscape studies for clusters and biomolecules with an extensive list of references can be found in David Wales' book (Wales, 2005). It is also noteworthy that Berry and collaborators popularized the landscape paradigm through the Telluride Energy Landscape workshops that regularly take place in Telluride, Colorado, since 1984.

Since the past decade the landscape metaphor enjoys widespread use in other fields of physics and chemistry as well. "Synchronisation landscapes" are used to elucidate the properties and nonlinear dynamics of complex networks (Zhou, 2003; Nishikawa and Motter, 2010). Rabitz and co-workers introduced the notion of the "Quantum Control Landscape" for the analysis of quantum-mechanical observables as a function of controls (Chakrabarti and Rabitz, 2007). Topological properties of such landscapes are studied in (Hsieh et al., 2008, 2009). The relationship between the structure of quantum control landscapes and optimization complexity is considered in (Moore et al., 2008). We explore the general relationship between landscapes and optimization in the next section.

## 2.3 Landscapes in optimization

Thus far, we have seen fundamental landscape instances in the natural sciences. The beauty of the landscape concept is that these "natural" landscapes are a subset of the more general class of landscapes that arise from distinct combinations of optimization problems *and* search algorithms. For example, the domain of the genetic fitness landscapes is inherently discrete as we consider strings of length $n$ over the finite alphabet $\{A, C, G, T\}$. Together with a fitness assignment for each string, finding the combination of letters that maximizes or minimizes the fitness defines a *combinatorial optimization problem*. Adding a distance or neighborhood relation between strings leads to a *combinatorial optimization landscape* (Reidys and Stadler, 2002). We have previously introduced some notions of distance, such as the Hamming distance associated with single-site changes of letters or distances based on abstract move sets. For a general optimization problem these move sets can be associated with iterations of an optimization algorithm applied to the specific problem instance. However, before developing this idea

further we have to raise an important question: When and why is the landscape perspective valuable for optimization problems? The answer to this question is intimately related to the computational complexity of the studied problem class and the employed algorithm.

### *Computational complexity and algorithm classes*

Over the past decades, computer scientists and mathematicians have developed a formidable classification scheme for optimization problems. This scheme is related to the resources, both in terms of memory and computational time, an algorithm needs to solve all problems in a specific class. This is the realm of computational complexity and algorithm classes. Consider for instance one of the best-studied combinatorial optimization problems, the *Traveling Salesman Problem* (TSP): Given a list of $n$ cities and their pairwise distances, the task of the "salesman" is to find a the shortest possible tour that visits each city exactly once and returns to the city the tour started from. TSP belongs to the complexity class of *NP-complete problems*. The acronym NP refers to **N**ondeterministic **P**olynomial time. The characteristic of NP-complete problems is the following: Any *given* solution to such a problem can be quickly verified, but there is no known efficient way to locate a solution in the first place. Efficiency here refers to the required resources as a function of the problem (or input) size $n$. Indeed, the time required to find the optimal solution using any currently known algorithm must grow faster than any polynomial in $n$. Simplified versions of the protein folding problem are also in the class of NP-complete problems (Hart and Istrail, 1997). A ubiquitous technique in computational complexity is the method of reduction. Proving that a certain problem is NP-complete can be done by first showing that it is NP and then transforming (reducing) it to a problem that is already known to be NP-complete (such as TSP). Hence, algorithms that tackle TSP can also be used for other problems. Belonging to the NP class does, however, not imply that *any* instance of the problem is hard, rather that there *exist* hard instances. Two approaches have been developed to deal with NP-complete problems: *approximation algorithms* and *heuristics*. Approximation algorithms are problem-specific methods that find sub-optimal solutions in polynomial time with *provable* solution quality. This means that the found approximations are optimal up to a constant factor, for instance within 10% of the optimal solution. Approximation algorithms are sometimes also used when exact polynomial-time algorithms are known, but are still too expensive for a given input size. Heuristics, on the other hand, are computational methods that can often be applied to a wider range of problems at the expense of providing no guarantees about the goodness of the solutions found. Heuristics often rely on iteratively improving intermediate candidate solutions until some stopping criteria are met. The methods we present in Chapter 4 belong to this class. Heuristics are applied whenever exact or approximation algorithms are too expensive or are not known. In such situations, the landscape paradigm can provide information that explains the success of a heuristic or guides the design of effective algorithms for a large number of problem instances.

## *Landscapes of NP-complete problems*

Probably the best-studied subclass of TSP is the *symmetric TSP*, where the distances between cities are symmetric, i.e., traveling from city A to B takes the same amount of time as traveling from B to A. The first landscape analysis of symmetric TSP appeared in the Operations Research community in the early 1990's through the works of Kenneth D. Boese and co-workers (Boese et al., 1994; Boese, 1995). Studying specific landscape characteristics enabled them to develop a new, effective stochastic multi-start heuristic for solving certain instances of TSP (Boese et al., 1994). The principal idea behind their contributions was the analysis of the relationship between the cost of a tour and distance to nearby local and global minima. A large sample of sub-optimal tours $t_k$ was generated on a well-known $n = 532$ cities instance (ATT532) for which the optimal tour had been solved using a branch-and-cut algorithm (Padberg and Rinaldi, 1987). The sub-optimal solutions were obtained by local search methods that iteratively improve candidate solutions through the application of specific operators. Most of these operators or move sets were of the k-opt type (Croes, 1958; Lin and Kernighan, 1973). The simplest k-opt variant is 2-opt (Croes, 1958), which deletes two nonadjacent edges of the current tour and then reconnects the two resulting paths into a new tour. This specific move set induces a distance on the landscape, for instance the minimum number of 2-opts needed to transform $t_i$ into $t_j$. Boese and co-workers simplifed this by using as a distance the number $n$ of shared edges in tours $t_i$ and $t_j$. This number approximates the minimal number of 2-opt moves between any two tours by a factor of at most two (Boese et al., 1994). The cost of a tour was defined in the usual way as the length of the total tour. The landscape analysis of the TSP instance revealed two surprising results: (i) There is a

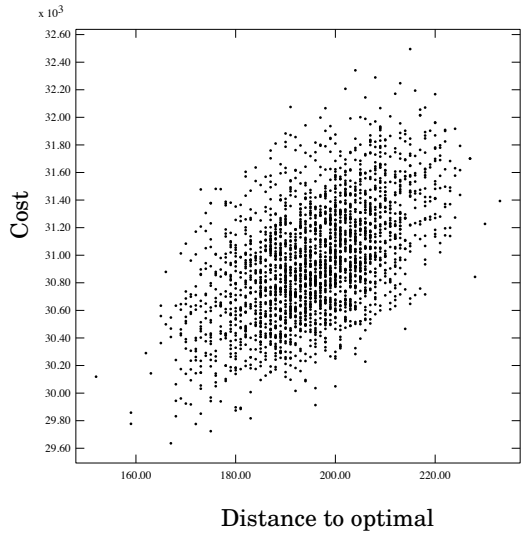



Figure 2.9: 2500 random 2-opt local minima for ATT532. Tour cost is plotted vs. distance to global minimum (from (Boese, 1995))

correlation between the distance to the optimal solution and the cost of a tour (see Fig. 2.9),

and (ii) there is a strong correlation between the mean distance to other solutions and the cost of a tour, independent of the employed operators. These observations led the authors to formulate the *big valley hypothesis* for the TSP landscape: Low-cost solutions are located in a single valley around the optimal solution. The data also inspired the authors to design an adaptive multi-start strategy that exploits this big-valley structure to obtain a considerable performance increase compared to standard multi-start strategies. The big-valley structure is synonymous in our terminology to a single-funnel landscape. An alternative expression is the notion of a *globally convex* landscape, a term that has first been introduced by Hu and co-workers (Hu et al., 1989) in an attempt to generalize the notion of convexity from functional analysis. Correlation analysis between cost (or fitness) and distance between solutions has also been essential in the study of Kauffman's NK model as mentioned earlier. Kauffman himself provided the first cost-distance plots in (Kauffman, 1993). The fitness function of the general NK model is defined as:

$$f_{\mathrm{NK}}(s) = \sum_{i=1}^{n} f_i\left(s_i, \mathcal{N}(s_i)\right) \ . \tag{2.7}$$

The function $f_{\mathrm{NK}}$ assigns real values to binary strings $s$ of fixed length $n > 0$ and $s = (s_1, \ldots, s_n) \in \{0,1\}^n$. The total fitness $f_{\mathrm{NK}}$ is the sum of $n$ local fitness functions $f_i$. Each local fitness function depends on the main variable $s_i$ and its neighborhood $\mathcal{N}(s_i)$, that specifies $k$ positions in the string. For a given neighborhood structure, the local fitness function $f_i$ is determined by a fitness lookup table that specifies the function value $f_i$ for each of the $2^{k+1}$ possible assignments to the variables $s_i$ and $\mathcal{N}(s_i)$. The main parameters of the NK model are $n$ and $k$. They define the size of the search space and the number of neighbors. The beauty of the NK models lies in the fact that, for a fixed $n$, the parameter $k$ can be used to tune the landscape from a simple additive function ($k = 0$) to a purely random landscape ($k = N - 1$). The parameter $k$ reflects the interaction strength (epistasis) between different sites. Weinberger proved that finding the global minimum of $f_{\mathrm{NK}}$ is NP complete for $k \geq 2$ (Weinberger, 1996). Nonetheless, Kauffman showed that for a specific instance of the NK model with $n = 96$, there is still considerable correlation between the fitness of local optima and their mutual Hamming distance for $k = 2$. Using this information, an adaptive local search heuristic might still be able to find the global minimum efficiently. The correlation structure is, however, lost for $k = 4$ (Kauffman, 1993), limiting the success of local search procedures on this instance. A more recent study on the interplay between search operators and NK landscapes can be found in (Merz, 2005).

Correlation between fitness and distance as a general measure of problem difficulty has been popularized in (Jones and Forrest, 1995). They called this measure *Fitness-distance correlation* (FDC) and successfully showed that it can explain the performance of genetic algorithms on a set of combinatorial benchmark problems. Ever since, FDC analysis has been applied to many landscapes arising from combinations of search heuristics and NP-complete problems, such as the Graph Bi-Partitioning problem (Merz and Freisleben, 1998) or the unconstrained binary Quadratic Assignment problem (Merz, 2004). An interesting approach relating landscape topology and design of search algorithms has been proposed by Ikeda and Kobayashi for the Job-shop Scheduling Problem (JSP) (Ikeda and Kobayashi, 2000). The job-shop scheduling problem consists of optimally assigning jobs to resources at particular times. The most

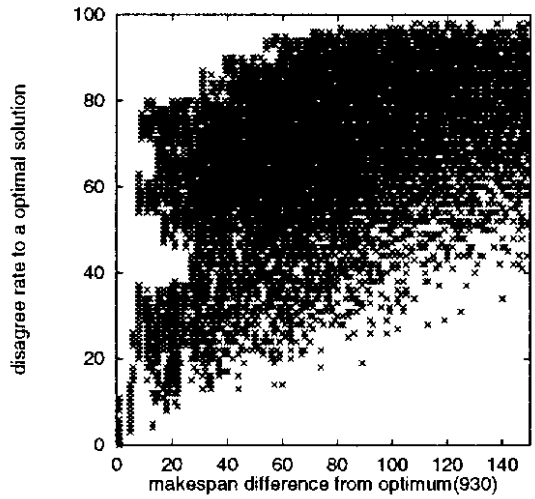



Figure 2.10: Relation between the disagreement rate of job order (distance) to the optimum (y-axis) and the makespan (fitness) for FT10 (from (Ikeda and Kobayashi, 2000)).

basic version is as follows: Given $n$ jobs $j_1, j_2, \ldots, j_n$ of varying sizes. All jobs need to be scheduled on $m$ identical machines, such as to minimize the total length of the schedule, or the make span. There are many problem variations, including constraints on the order of the jobs or online vs. offline scenarios. Ikeda and Kobayashi realized that the "big valley structure" hypothesis does not apply to many test problems of JSP, including the problems in the well-known Fisher-Thompson (FT) library (Fisher and Thompson, 1963). An FDC plot of instance FT10 is shown in Fig. 2.10. It indicates that the landscape has multiple near-optimal solutions that are widely separated in solution space. At the same time, empirical evidence from several studies suggested that local search heuristics including genetic algorithms, notoriously fail on these instances. In order to qualitatively explain both the observed problem topology and the reduced algorithm performance Ikeda and Kobayashi extended the "big valley structure" hypothesis to a double-funnel topology, called the *UV-structure*. UV landscapes consist of a $U$-shaped valley which is broad and shallow and a $V$-valley which is narrow and deep that contains the global minimum. Local search methods or population-based heuristics are likely to explore the $U$-shaped valley, hence failing to find the minimum in the $V - shaped$. In order to remedy this behavior of standard heuristics on multi-funnel topologies, Ikeda and Kobayashi proposed the *Innately Split Model* (ISM). The ISM starts local searches in several groups that are initially spread across the landscape domain. When searches come close to each other, one of them is removed and randomly restarted somewhere else. This simple strategy increases the probability of exploring the V-shaped valley, while avoiding unnecessary searches in the U-shaped valley. Applying this model to JSP instances drastically increased the search performance (Ikeda and Kobayashi, 2000).

The combinatorial landscape studies presented so far have all been empirical in nature. The analysis of possibly representative problem instances led to the development of novel heuris-

tics that showed improved performance on a wider class of problem instances. A different, more rigorous approach are autocorrelation analyses of landscapes by Edward D. Weinberger (Weinberger, 1990). Weinberger's fundamental contribution was to make the notion of landscape ruggedness precise. His method is based on generating random walks on the landscape and estimating the autocorrelation function between fitness and walk length. A formal definition of this autocorrelation can be found in Eq. (3.10). Weinberger provided both a first mathematical treatise of the topic and numerical simulations on NK fitness landscapes. Although mostly focused on biological fitness landscapes, he also commented on the hardness of general optimization problems. For NK landscapes, his autocorrelation framework also confirmed the sharp increase in problem difficulty from $k = 2$ and $k = 4$, as previously discussed. In a series of papers throughout the past two decades Peter F. Stadler and coworkers extended the idea of autocorrelation analysis, culminating in a general *algebraic* or *spectral theory* of landscapes (see e.g. (Schuster and Stadler, 1994; Stadler, 1995, 1996; Reidys and Stadler, 2002)). Stadler investigated both classical combinatorial optimization landscapes and biological fitness landscapes, mostly related to RNA evolution. Within his theory, he analytically derived autocorrelation functions for many NP-complete problems. He found that specific combinations of NP problems and neighborhood definitions for a random walk lead to so-called *elementary landscapes* with exponential autocorrelation function. A comprehensive list of known autocorrelation coefficients, a derived quantity of the autocorrelation function, for combinatorial landscapes can be found in (Angel and Zissimopoulos, 2000).

Spectral landscape theory is to date the most rigorous approach toward a better understanding of combinatorial optimization landscapes. An analog for continuous black-box landscapes (see Def. 2.2) does not exist, and will probably never exist. This is due to the generality of the landscape definition and the lack of underlying assumptions. Nonetheless, inspiration can be drawn from the wealth of works presented in the areas of science and combinatorial optimization, and some techniques may be transferred to the field of continuous black-box optimization.

### 2.3.1 Continuous black-box landscapes and their impact on optimization

Continuous black-box optimization problems are ubiquitous in science and engineering. They occur in many practical applications ranging from simple parameter identification in data model fitting to intrinsic design-parameter optimization in complex technical systems. The diversity of these real-world problems hampers a clean classification of problem structure and complexity. We advocate that the landscape perspective offers a way to establish a more refined analysis of continuous black-box optimization problems. It is conceivable that "archetypal" landscape topologies are also present in many instances of black-box problems. We summarize the key topologies we encountered so far in Fig. 2.11. The simplest topology is a convex (and hence single-funnel) structure (Fig. 2.11a). This landscape has only one minimum, which is the global one. If one knows in advance that both the landscape domain and the objective function are convex, there is a wealth of exact and efficient techniques for finding the global minimum. We refer to the excellent book of Boyd and Vandenberghe for an overview (Boyd and Vandenberghe, 2004). A globally convex single-funnel landscape topology (Fig. 2.11b) consists of a number of local minima that can be seen as high-frequency

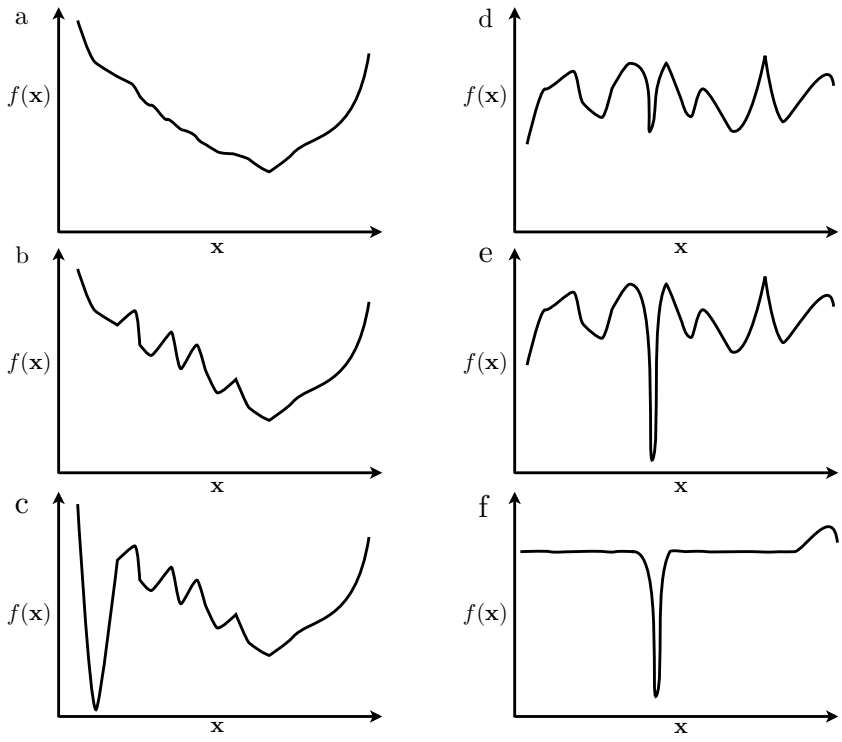



Figure 2.11: Sketches of archetypal landscape topologies. a. Convex single-funnel landscape. b. Globally convex single-funnel landscape. c. Double-funnel landscape with a broad sub-optimal funnel. d. Multi-modal landscape with minimum at the boundary and no funnel structure. e. Same as d but with a deep, needle-like minimum. f. Golf-course or needle-in-the-haystack topology with large regions of neutrality.

perturbations to an underlying convex structure (the big valley structure). Hu and co-workers (Hu et al., 1989) attempted to make this notion precise by establishing the "$\delta$-convexity" property of a function on convex domains. The idea of $\delta$-convexity is to allow non-convex variations of the function on a length scale $\delta$ that is small compared to the size of the domain. For *all* pairs of points separated by more than $\delta$, convexity must hold in the usual sense. For our purposes, however, this definition is not practical, as we comment on in the next section. Another archetypal landscape structure is the double-funnel topology (Fig. 2.11c) that we have seen in the case of Clark's folding/aggregation energy landscape and the UV-structure of the JSP instances. Whenever the funnel that contains the global minimum covers a much smaller domain than the other funnels, it poses considerable challenges for the success of local search heuristics, since most searches are drawn toward the bottom of the sub-optimal funnel. In the black-box optimization community such a landscape is also called *deceiving*. The double-funnel case is the simplest instance of the class of multi-funnel landscapes. Fig. 2.11d and e show multi-modal landscapes with no global funnel structure. The notoriously hard golf-course landscape or needle-in-the-haystack topology is depicted in Fig. 2.11f, where large

flat regions surround a single narrow minimum.

Despite the tremendous number of novel black-box optimization heuristics published in the past two decades, limited attention has been paid to the question what global topology a certain problem instance has, how to measure it, and how success or failure of a certain algorithm can be related to landscape topology. Few notable exceptions exist. Hansen and Kern (Hansen and Kern, 2004) pointed out that for CMA-ES "a strong asymmetry of the underlying function jeopardizes a successful detection and can lead to a failure." However, "if the local optima can be interpreted as perturbations of an underlying unimodal function", CMA-ES performs well. Lunacek and co-workers investigated in their paper "The impact of global structure on search" (Lunacek et al., 2008) the performance of heuristics on double-funnel landscapes. Kobayashi and co-workers explicitly took into account landscape topology for both algorithm design and interpretation of performance (see for instance (Ikeda and Kobayashi, 2000; Sakuma and Kobayashi, 2001)). Wang and Li (Wang and Li, 2008) generalized the NK model from the discrete to the continuous domain using the concept of linkage function that defines the interaction (epistasis) between variables. They introduced a number of continuous NK models with tunable epistasis strength and tested the capabilities of several search heuristics on these landscapes. They were also the first to apply the FDC measure for continuous fitness landscapes.

Given the lack of a common framework for continuous landscape analysis, we dedicate Chapter 3 to this topic. We try to extend the present approaches within a statistical sampling framework. Prior to this analysis we introduce traditional black-box test functions and a benchmark suite that serve as test beds throughout this thesis.

### 2.3.2 Classical black-box optimization problem landscapes

Besides countless real-world applications of black-box heuristics, the design of novel algorithms has traditionally been accompanied by numerical simulations on sets of benchmark functions. Over the past 50 years, a surprisingly limited number of benchmark problems has formed the common basis for algorithmic performance evaluation. These function are often named after the author who defined or used them for the first time. In bio-inspired continuos optimization, prominent examples are the Rastrigin, Rosenbrock, Ackley and Griewank function along with the test sets provided by Kenneth de De Jong (De Jong, 1975) and Hans-Paul Schwefel (Schwefel, 1993). Many of these functions have specific features that allow drawing conclusions about the search and convergence behavior of the tested search strategies. In the following we introduce the benchmark problems used in this thesis.

#### *Sphere function*

The sphere function is the prototypical quadratic function that is fundamental both for theoretical and empirical convergence studies of black-box algorithms. This separable function is

defined as:

$$f_{\text{Sphere}}(\mathbf{x}) = \sum_{i=1}^{n} x_i^2 \,. \tag{2.8}$$

The global minimum is at the origin $\mathbf{0}$ with $f_{\text{Sphere}}(\mathbf{0}) = 0$. For practical purposes, search is often restricted to $\mathbf{x} \in [-5, 5]^n$. For the sphere function, an impressive body of theoretical work exists with progress rates and convergence proofs of Evolution Strategies. We refer to the excellent book of Hans-Georg Beyer for an overview (Beyer, 2001).

### *Rosenbrock function*

Another standard test function is the generalized Rosenbrock (valley) or banana function, Fig. 2.12, right):

$$f_{\text{Rosen}}(\mathbf{x}) = \sum_{i=1}^{n-1} \left(100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right) \,. \tag{2.9}$$

The global minimum is at $\mathbf{1}$ with $f_{\text{Rosen}}(\mathbf{1}) = 0$. Search is usually constrained to $\mathbf{x} \in [-2, 2]^n$. Rosenbrock's function is multi-modal for $n > 3$, and it exhibits the interesting topology of a curved valley. On a global length scale ($\|x_i\| > 1$), the first summand dominates and attracts most search heuristics toward the origin. On smaller length scales ($\|x_i\| \ll 1$), however, the second term dominates and forms a bent parabolic valley that leads from the origin to the global minimum at $\mathbf{1}$. On this function it is therefore favorable to constantly reorient the search direction along the valley.

### *Rastrigin function*

The Rastrigin function (Fig. 2.12 left) can be considered as the prototypical multi-modal function with an underlying globally convex (or even more, quadratic) topology. It is defined as:

$$f_{\text{Rast}}(\mathbf{x}) = A\,n + \sum_{i=1}^{n} \left(x_i^2 - A\cos\omega x_i\right) \,. \tag{2.10}$$

Standard settings for amplitude and frequency are $A = 10$ and $\omega = 2\pi$. The global minimum is at $\mathbf{0}$ with $f_{\text{Rast}}(\mathbf{0}) = 0$, and the domain is restricted to $\mathbf{x} \in [-5, 5]^n$. Far away from the minimum the quadratic term dominates and the basin depths decrease. Close to the minimum the cosine term renders the landscape highly rugged with basins of depth $\approx A$. Within the prescribed domain the Rastrigin function contains $11^n$ basins. The interior basins all have unit volume, the basins at the boundary have a location-dependent volume. For instance, at an n-dimensional corner the basin size is $0.25^n$. Because the amplitude $A$ is considerably smaller than the total scale of function values, and each minimum has several neighboring minima (under Euclidean distance) that are lower in fitness, we consider the Rastrigin function a single-funnel landscape.

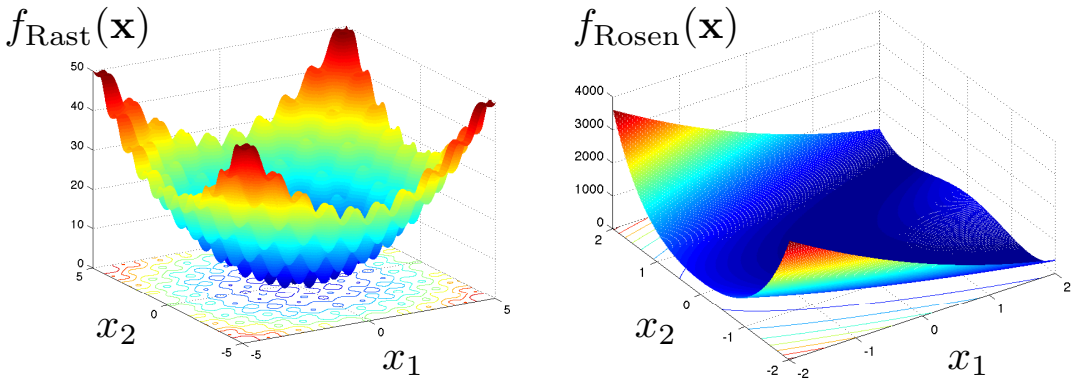



Figure 2.12: Rastrigin (left) and Rosenbrock (right) function in 2D

### *Lunacek's double-funnel functions*

Lunacek and co-workers (Lunacek et al., 2008) introduced two functions that have a parametrizable double-funnel topology. The first function is the Double Sphere $f_{\text{DS}}$:

$$f_{\text{DS}}(\mathbf{x}) = \min\left(\sum_{i=1}^{n}(x_i-\mu_1)^2\,,\, d\,n + s\sum_{i=1}^{n}(x_i-\mu_2)^2\right) \tag{2.11}$$

with $\mu_1 = 2.5$ and $\mu_2 = -2.5$. The domain is restricted to $\mathbf{x} \in [-5,5]^n$. The landscape can be tuned with the parameters $s$ and $d$, where $s$ controls the size and $d$ the depth of the second funnel. Lunacek and co-workers further proposed to consider $s$ the primary control variable for tests and to ensure that the two funnels always intersect at the origin. For any fixed $d$, this can be achieved by choosing $\mu_2 = -\sqrt{(\mu_1^2 - d)\,/s}$. The second function is constructed from

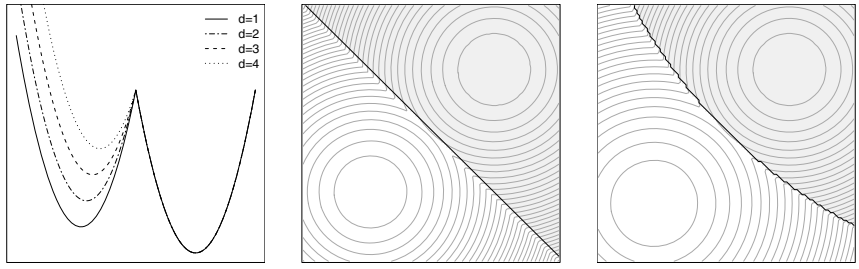



Figure 2.13: The influence of $d$ and $s$ on the double-sphere function. Increasing $d$ creates more difference between the basin depths (left). When $s = 0$ (middle), the two basins have the same size. Decreasing $s$ creates a larger sub-optimal basin (right) (from (Lunacek et al., 2008)).

$f_{\mathrm{DS}}(x)$ and a Rastrigin function as:

$$f_{\mathrm{DR}}(\mathbf{x}) = f_{\mathrm{DS}}(x) + 10 \sum_{i=1}^{n} \left(1 - \cos 2\pi \left(x_i - \mu_1\right)\right) . \tag{2.12}$$

This Double-Rastrigin function $f_{\mathrm{DR}}(x)$ is a prototypical rugged double-funnel landscape with similar basin size distribution as the original Rastrigin function.

### *Kjellström's function*

This multi-modal function has been proposed by (Kjellström and Taxen, 1992). We hence propose to call this function Kjellström's function $f_{\mathrm{Kjell}}$. It is defined as:

$$f_{\mathrm{Kjell}}(\mathbf{x}) = \prod_{i=1}^{n}(1 + h(x_i)), \quad h(x_i) = 0.01 \sum_{j=1}^{5}[\cos(jx_i + b_j)] , \tag{2.13}$$

with $\mathbf{b} = [b_1, \ldots, b_5] = [1.982, 5.720, 1.621, 0.823, 3.222]$ and $\mathbf{x} \in [0, 2\pi]^n$. In the original publication the location of the global minimum $\mathbf{x}_{\min}$ is said to be roughly at $x_{i\,\min} = 2.34, i = 1, \ldots, n$. We numerically determined the more accurate value of $x_{i\,\min} = 2.34861543, i = 1, \ldots, n$ with the minimum value $f_{\mathrm{Kjell}}(\mathbf{x}_{\min}) \approx 0.96916908^n$. Figure 2.14 depicts the one-dimensional Kjellström function. In 1D, $f_{\mathrm{Kjell}}$ has 5 minima. The global maximum $\mathbf{x}_{\max}$ (located at a value slightly larger than $\mathbf{x} = \pi$) divides the search space into two parts: The region $\mathbf{x} < \mathbf{x}_{\max}$ covers a bit more than half of the space (solid green bar in Fig. 2.14). This region contains, on average, lower function values than the other region, which simplifies searching for the global minimum $x_{\min}$. The basin sizes vary from $\approx 0.5$ to $\approx 1.5$. The $n$-dimensional $f_{\mathrm{Kjell}}$ is the Cartesian product of $n$ 1D functions. Kjellström's function is, hence, a separable multi-modal function with $5^n$ basins and no funneled topology.

### *Black-box optimization benchmark suites*

A common shortcoming of many empirical optimization studies using classical benchmark functions is the lack of a standard protocol of how to perform the numerical simulations. Each publication usually considers its own subset of test functions, number of allowed function evaluations, number of repetitions of the experiments, dimensionality of the problems, and performance measures. This makes it impossible to compare results across publications. One of the earliest attempts to standardize these benchmarks was a contest on numerical optimization at the International Conference of Evolutionary Computation in 1996 (Bersini et al., 1996). Whitley and co-workers (Whitley et al., 1995) developed guidelines for the design of meaningful test suites and showed that standard test functions do not follow these guidelines. Three key requirements were proposed: Test suites should contain (i) landscapes that are resistant to "hill-climbing" methods, (ii) non-linear, non-separable landscapes, and (iii) non-separable and scalable landscapes. Hill-climbing methods are iterative local search methods that choose strictly improving steps in order to reach the next optimum. Highly

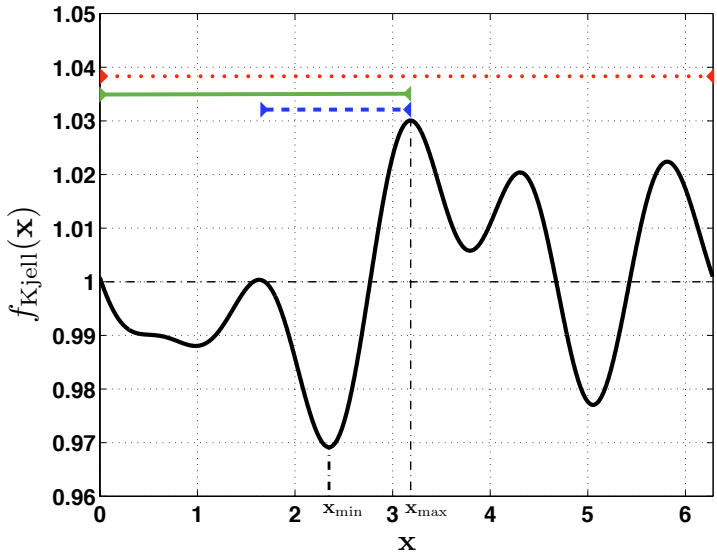



Figure 2.14: The multi-modal function $f_{\text{Kjell}}$ in 1D. The global minimum $\mathbf{x}_{\text{min}}$ is contained in a locally convex region (blue dashed bar) that belongs to a sub-region of the space (solid green bar) that is slightly larger than $\pi$. The global maximum $\mathbf{x}_{\text{max}}$ separates this region from the right part of the space. The red dotted bar spans the full search space of length $2\pi$

multi-modal landscapes are resistant to such methods because a hill climber possibly needs to explore *all* modes for successfully finding the global minimum. From our previous examples we see that the property of simultaneous non-linearity and non-separability is not satisfied by the Sphere, the Rastrigin and the Kjellström functions. There is no non-linear interaction *between* the variables. Hence, global optimization can be reduced to $n$ one-dimensional search problems. Non-separability can easily be achieved by rotating the landscape domain. The property of scalability is concerned with the computational cost of evaluating the function for increasing dimensionality. Consider the problem $f_1(x_1, x_2)$ where both variables interact with each other. An $n$-dimensional generalization with a linear scaling behavior can be constructed through expanded functions where only pairs of variables interact, e.g., $f_n(x_1, \ldots, x_n) = f_1(x_1, x_2) + \ldots + f_1(x_{n-1}, x_n)$. The Rosenbrock function is an example of a scalable function.

It was, however, not before the 2005 IEEE Congress on Evolutionary Computation that a comprehensive and well balanced suite of test functions was agreed on. This IEEE CEC 2005 benchmarks considered many of the above criteria (Suganthan et al., 2005). The test suite has been designed by experts for the IEEE CEC 2005 Special Session on Real-Parameter Optimization. It is intended to define a standard benchmark for real-parameter optimization algorithms, along with standardized evaluation criteria and testing procedures, thus allowing

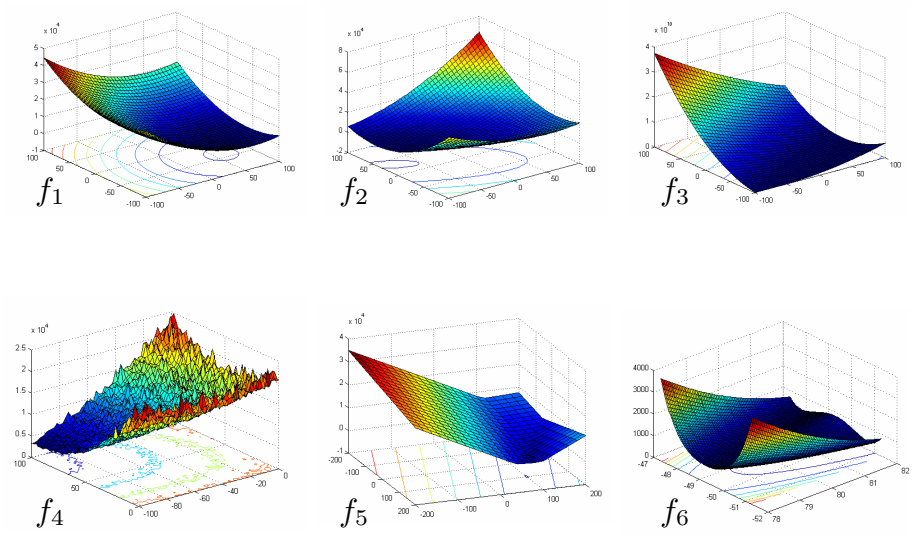



Figure 2.15: 2D landscapes of CEC functions $f_1$–$f_6$

performance comparisons of different optimization algorithms across publications. The suite consists of 25 functions with different properties. The names of the functions are listed in Table 2.1. Functions f1 to f5 are unimodal, f6 to f12 are basic multi-modal. Functions f13 is a expanded function consisting of two different functions. Function f14 is an expanded function consisting of two-dimensional Scaffer's F6 functions with different parameterizations. Functions f15 to f25 are composite test functions that are formed by superposition of more than two standard test functions. From a landscape perspective we consider f11–f13 and f15–f25 multi-funnel instances (see Fig. 2.15 to Fig. 2.17 for 2D versions of all CEC landscapes). In order to prevent exploitation of search space symmetry, all problems are shifted and many of them are rotated. This means that the global minimum is never located in the center of the search domain. Moreover, the global minimum of each function is different from the common zero value. Rotation of the search space makes almost all problems non-separable. Functions f4 and f17 are corrupted by addition of a noise term that vanishes at the global minimum. All problems are box-constrained, except functions f7 and f25, which are unconstrained. An advantageous feature of the IEEE CEC 2005 test suite is the existence of comparison groups of similar functions, allowing sensitivity tests of search algorithms with respect to changing features of the problem. The functions f1–f3 are all quadratic functions with different condition numbers of the Hessian $H(\mathbf{x}_{\min})$. The function f4 is the same as f2 with an additional noise term. Function f10 is a rotated version of f9 that is essentially the Rastrigin function with shifted global minimum. We refer to the 50-page technical report of the test

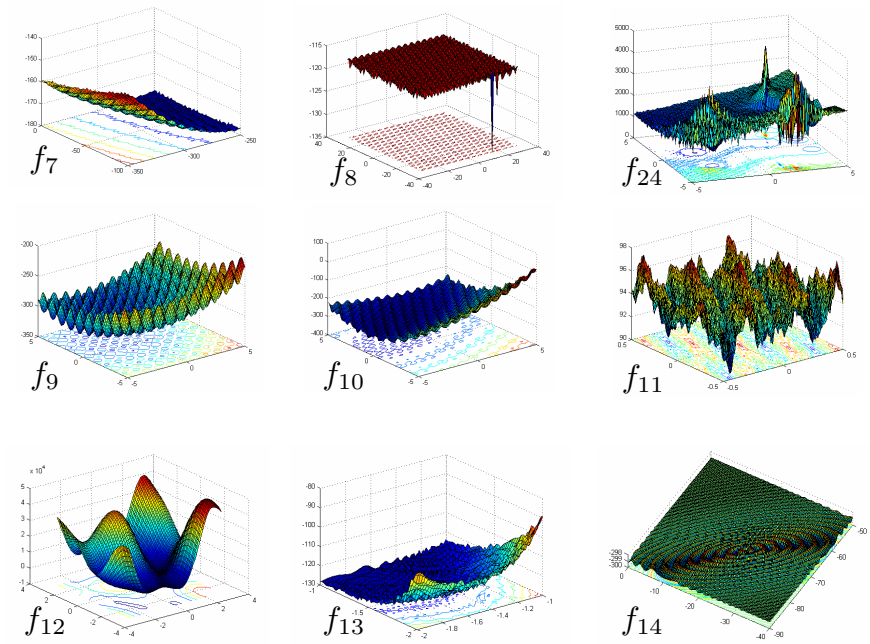



Figure 2.16: 2D landscapes of CEC functions $f_7$–$f_{14}$ and $f_{24}$

suite (Suganthan et al., 2005) for a full list of comparison groups and exact function definitions.

An important issue of the suite is the experimental protocol. All 25 functions are supposed to be evaluated 25 times for $n = 10, 30, 50$ dimensions. The allowed budget of function evaluations (FES) is restricted to MAX_FES= $10^4\, n$ for each run. This reflects the limited resources often encountered in real-world applications, where an acceptable solution should be found within a restricted number of black-box evaluations. The benchmark settings are summarized in Table 2.2. Furthermore, the benchmark suite specifies the level of accuracy $\epsilon$ for the optimal solutions. An algorithm is considered to have solved a certain problem if it reaches an objective value $f(\mathbf{x}) < f(\mathbf{x}_{\min}) + \epsilon$ (see Table 2.3). Since 2005, a large number of algorithms has been tested on this benchmark suite. It was thus the natural choice for our work, which started in 2007. We note, however, that recently a more flexible test bed has been introduced: the COCO (Comparing Continuous Optimisers) platform for Black-Box Optimisation Benchmarking (BBOB), presented at two GECCO workshops in 2009 and 2010. In COCO/BBOB, both noise-free and noisy test functions are provided, including Lunacek's Double-Rastrigin landscape and a function created by Gallagher's landscape generator (Gallagher and Yuan, 2006). The COCO platform moreover includes scripts for automatic post-processing and presentation of the results in a unified manner. More details can be found at `http://coco.gforge.inria.fr/doku.php`. Testing our techniques and algorithms within the COCO platform will be a topic of future research.

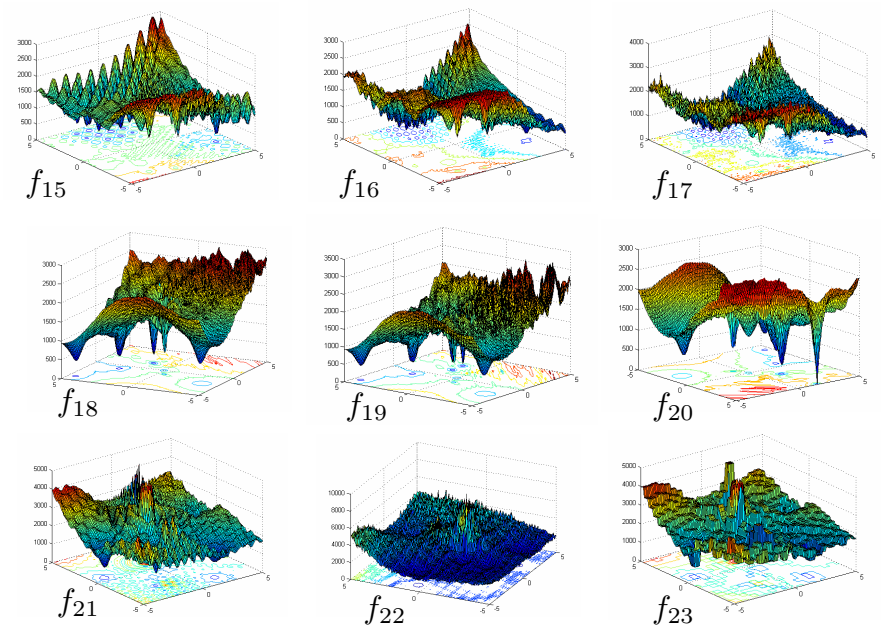



Figure 2.17: 2D landscapes of CEC functions $f_{15}$–$f_{23}$

All available standardized continuous black-box benchmark test cases are based on synthetic test functions. A benchmark test suite that includes real-world problems from science and engineering is thus far not available. It is, however, conceivable that algorithms that perform well on synthetic problems may show reduced performance in real-world applications. Recent investigations on space mission design problems support this hypothesis (Vasile, 2010). In Chapter 6 we therefore propose the energy landscapes of certain atomic cluster instances as real-world optimization benchmarks. Following the design principles of the IEEE CEC 2005 benchmark suite, we introduce a diverse set of problems along with a standardized experimental protocol. We argue that these benchmarks should be included in future benchmark studies in order to test the effectiveness and generality of continuous black-box optimizers.

| Function | Name |
|---|---|
| $f_1$ | Shifted Sphere Function |
| $f_2$ | Shifted Schwefel's Problem 1.2 |
| $f_3$ | Shifted Rotated High Conditioned Elliptic Function |
| $f_4$ | Shifted Schwefel's Problem 1.2 with Noise in Fitness |
| $f_5$ | Schwefel's Problem 2.6 with Global Optimum on Bounds |
| $f_6$ | Shifted Rosenbrock Function |
| $f_7$ | Shifted Rotated Griewank Function without Bounds |
| $f_8$ | Shifted Rotated Ackley Function with Global Optimum on Bounds |
| $f_9$ | Shifted Rastrigin Function |
| $f_{10}$ | Shifted Rotated Rastrigin Function |
| $f_{11}$ | Shifted Rotated Weierstrass Function |
| $f_{12}$ | Schwefel's Problem 2.13 |
| $f_{13}$ | Expanded Extended Griewank plus Rosenbrock Function (F8F2) |
| $f_{14}$ | Shifted Rotated Expanded Scaffer's F6 |
| $f_{15}$ | Hybrid Composition Function 1 |
| $f_{16}$ | Rotated Hybrid Composition Function 1 |
| $f_{17}$ | Rotated Hybrid Composition Function 1 with Noise in Fitness |
| $f_{18}$ | Rotated Hybrid Composition Function 2 |
| $f_{19}$ | Rotated Hybrid Composition Function 2 with a Narrow Basin for the Global Optimum |
| $f_{20}$ | Rotated Hybrid Composition Function 2 with the Global Optimum on the Bounds |
| $f_{21}$ | Rotated Hybrid Composition Function 3 |
| $f_{22}$ | Rotated Hybrid Composition Function 3 with High Condition Number Matrix |
| $f_{23}$ | Non-Continuous Rotated Hybrid Composition Function 3 |
| $f_{24}$ | Rotated Hybrid Composition Function 4 |
| $f_{25}$ | Rotated Hybrid Composition Function 4 without Bounds |

Table 2.1: Names of the test functions according to the CEC 2005 test suite (Suganthan et al., 2005).

| | |
|---|---|
| Problems | $f_1 - f_{25}$ |
| Runs per problem | 25 |
| Dimensionality $n$ | 10, 30, 50 |
| MAX_FES | $10^4 \cdot n$ |
| Termination | If FES = MAX_FES or $f_{\text{err}}(x) \leq 10^{-8}$ |
| Initialization | Uniform random position |

Table 2.2: Benchmark settings according to the CEC 2005 test suite (Suganthan et al., 2005).

| **Function** | $f_i(\mathbf{x}_{\min}) + \epsilon$ | **Function** | $f_i(\mathbf{x}_{\min}) + \epsilon$ |
|---|---|---|---|
| $f_1$ | $-450 + 1e{-}6$ | $f_{14}$ | $-300 + 1e{-}2$ |
| $f_2$ | $-450 + 1e{-}6$ | $f_{15}$ | $120 + 1e{-}2$ |
| $f_3$ | $-450 + 1e{-}6$ | $f_{16}$ | $120 + 1e{-}2$ |
| $f_4$ | $-450 + 1e{-}6$ | $f_{17}$ | $120 + 1e{-}1$ |
| $f_5$ | $-310 + 1e{-}6$ | $f_{18}$ | $10 + 1e{-}1$ |
| $f_6$ | $390 + 1e{-}2$ | $f_{19}$ | $10 + 1e{-}1$ |
| $f_7$ | $-180 + 1e{-}2$ | $f_{20}$ | $10 + 1e{-}1$ |
| $f_8$ | $-140 + 1e{-}2$ | $f_{21}$ | $360 + 1e{-}1$ |
| $f_9$ | $-330 + 1e{-}2$ | $f_{22}$ | $360 + 1e{-}1$ |
| $f_{10}$ | $-330 + 1e{-}2$ | $f_{23}$ | $360 + 1e{-}1$ |
| $f_{11}$ | $90 + 1e{-}2$ | $f_{24}$ | $260 + 1e{-}1$ |
| $f_{12}$ | $-460 + 1e{-}2$ | $f_{25}$ | $260 + 1e{-}1$ |
| $f_{13}$ | $-130 + 1e{-}2$ | | |

Table 2.3: Fixed accuracy levels according to the CEC 2005 test suite (Suganthan et al., 2005).

# 3

# Characterization of Black-box Landscapes

> "Mmmm, ... free samples"
> Homer Simpson, in: The Simpsons, Lisa gets an "A", Episode no. 210, 1998

We propose to characterize continuous black-box landscapes within a statistical sampling framework. The presented methods only require evaluations of the black-box function. Depending on the specific sampling strategy, we provide statistical estimators that address the following aspects of landscapes: global landscape topology, separability or variable epistasis, and landscape ruggedness. We focus on estimators that are easy to implement, easy to interpret, and computationally efficient. We consider black-box landscapes $\mathcal{L}_{\mathrm{B}}$ defined by a triple $(\mathcal{X}, d_{\mathrm{E}}, f)$ where $\mathcal{X}$ is box-constrained with $\mathcal{X} = [\mathbf{l}, \mathbf{u}] \subset \mathbb{R}^n$. The vectors $\mathbf{l}, \mathbf{u} \in \mathbb{R}^n$ define the lower and upper bounds. For unconstrained problems the techniques can be applied to any box-shaped region of interest of the landscape. Distances between points in the landscape domain are measured using the Euclidean distance $d_{\mathrm{E}}$. In order to test the discriminative power of the different techniques, they are applied to the full set of CEC 2005 benchmark functions. It is obvious that the accuracy of any of the presented methods will be limited by sample size. If the landscape exhibits fine structures below the sampling limit, they cannot be detected. Some of the presented methods have been introduced as "predictive measures of problem difficulty" in combinatorial optimization. We do not follow this notion here. In fact, it has been proven that, for certain problem classes, computing a general predictive measure is as hard as solving the problem itself (He et al., 2007). We rather envision the introduced statistical fingerprints as useful features based on which landscapes can be classified within a statistical learning framework.

## 3.1 Characterization of global topology

We present two techniques to characterize the global topology of a landscape: (i) Fitness-Distance Correlation (FDC) and (ii) Dispersion moments. Both techniques rely on a set of samples that is drawn uniformly at random from $[\mathbf{l}, \mathbf{u}]$.

### 3.1.1 Fitness-distance correlation

*Fitness-distance correlation* has been introduced by Boese (Boese et al., 1994) for the analysis of TSP and by Jones and Forrest (Jones and Forrest, 1995) as a "measure of problem difficulty" for the performance of genetic algorithms on combinatorial optimization problems. For continuous black-box problems, Wang and Li proposed this measure independently of us (Wang and Li, 2008). Given a uniform random sample $\mathbf{x}^{(j)} \in \mathcal{X}, j = 1, \ldots, S$ from the landscape, we evaluate the objective function at the sampled points and denote the values by $f^{(j)} \in \mathbb{R}, j = 1, \ldots, S$. In the original definition of FDC the location of the global minimum $\mathbf{x}_{\min}$ is assumed to be known *a priori*. While in a benchmark scenario this information is available, $\mathbf{x}_{\min}$ is approximated by $\tilde{\mathbf{x}}_{\min} = \arg\min_{\mathbf{x}^{(j)}} f(\mathbf{x}^{(j)}), j = 1, \ldots, S$ in the general case. Using the distances $d^{(j)} = d_{\mathrm{E}}(\mathbf{x}_{\min}, \mathbf{x}^{(j)})$ (or $d^{(j)} = d_{\mathrm{E}}(\tilde{\mathbf{x}}_{\min}, \mathbf{x}^{(j)})$, respectively), we define the fitness-distance correlation coefficient $r_{\mathrm{FD}}$:

$$r_{\mathrm{FD}} = \frac{c_{\mathrm{FD}}}{s_{\mathrm{F}} s_{\mathrm{D}}}\,, \tag{3.1}$$

with

$$c_{\mathrm{FD}} = \frac{1}{S}\sum_{j=1}^{S}(f^{(j)} - \bar{f})(d^{(j)} - \bar{d})\,, \tag{3.2}$$

and $\bar{f}$, $\bar{d}$, $s_{\mathrm{F}}$ and $s_{\mathrm{D}}$ the means and standard deviations of the fitness and distance samples, respectively. Although this measure is simple, it has been elucidative in a number of applications. The coefficient $r_{\mathrm{FD}}$ is expected to be near 1 for globally convex, single-funnel topologies and around 0 for needle-in-the-haystack problems and problems without any global structure. A negative value of $r_{\mathrm{FD}}$ indicates a "deceiving" landscape, i.e., a landscape on which a sampler or optimizer perceives larger objective function values closer to the minimum than farther away.

### 3.1.2 Function dispersion

*Function dispersion* has been introduced by Lunacek and Whitley (Lunacek and Whitley, 2006) in order to explain the search performance of CMA-ES. The dispersion of a black-box landscape is quantified by uniformly random samples $\mathbf{x}^{(j)} \in \mathcal{X}, j = 1, \ldots, S$ from the landscape and a target percentage $p$. The dispersion $dis^{\mathrm{m}}(s_b, S, f)$ of $f$ is calculated as the mean pairwise Euclidean distance between the best $s_b = p\,S$ samples. A given $p$ implicitly corresponds to a certain fitness threshold. The quantity of interest is the change in dispersion with decreasing $p$. In order to limit the number of distance computations, Lunacek and Whitely propose to fix the value $s_b$ to 100 and decrease $p$ (and hence the fitness threshold) by increasing the sample size $S$ until the FES budget is exhausted. The samples sizes

$S = 100 \cdot 2^0, \ldots, 100 \cdot 2^{12}$ have been used for landscapes in up to $n = 100$ dimensions in (Lunacek and Whitley, 2006). This corresponds to $p = 100\%, \ldots, 0.0024\%$. The mean dispersion difference $\Delta^{\mathrm{m}}_{\mathrm{dis}}(f) = dis^{\mathrm{m}}(100, 100 \cdot 2^{12}, f) - dis^{\mathrm{m}}(100, 100 \cdot 2^0, f)$ is used as an indicator to classify and compare functions. A negative value of $\Delta^{\mathrm{m}}_{\mathrm{dis}}(f)$ implies that the best fitness values of $f$ are localized in a small sub-region of the search space, a $\Delta^{\mathrm{m}}_{\mathrm{dis}}(f)$ value around 0 indicates that the best fitness values of $f$ are either spread over the entire search space *or* localized in distinct, remote funnels.

In order to be able to compare the dispersion values of objective functions with differently constrained search spaces, all sample points $\mathbf{x}^{(j)}$ are mapped from $[\mathbf{l}, \mathbf{u}]$ to the $[0, 1]^n$ hypercube and distances are evaluated in the hypercube. For $p = 100\%$ the estimator $dis^{\mathrm{m}}$ reduces to the hypercube line picking problem, i.e., to finding the average distance between two randomly chosen points in the cube. Closed-form solutions for this problem only exist for $n < 5$ (Weisstein, 2009).

The quantity $dis^{\mathrm{m}}(s_b, S, f)$ represents only the first moment of the distance distribution that can be monitored for a given $p$. We also analyzed the values of higher dispersion moments of the distributions such as variance, skewness, and kurtosis. We denote the variance of the distance distribution by $dis^{\mathrm{v}}(s_b, S, f)$ for a given $p$. This quantity is used to define the variance dispersion difference $\Delta^{\mathrm{v}}_{\mathrm{dis}}(f)$ analogously to the mean dispersion difference. In Section 3.4 we will present results both for $\Delta^{\mathrm{m}}_{\mathrm{dis}}(f)$ and $\Delta^{\mathrm{v}}_{\mathrm{dis}}(f)$.

## 3.2 Separability and variable importance

Detecting importance of variables and interactions among variables in complex models is a ubiquitous task in model building and analysis, commonly referred to as Sensitivity Analysis (Saltelli et al., 2000). In some cases, only a small subset of the variables or parameters of a model have significant effects on the system behavior. Likewise, some variables may be varied independently without affecting the influence of the others. An effective method to "screen" variable importance and interactions has been proposed by Morris (Morris, 1991). His method relies on a specific factorial design where only one parameter at a time (OAT) is changed. Consider the hypercube $[0, 1]^n$ as landscape domain covered by a regular, equidistant grid. Let $1/(g-1)$ be the smallest spacing between two parallel lines of the grid. The "level" of the grid is called $g$ and is assumed to be even. Let $\mathbf{x} = [x_1, \ldots, x_i, \ldots, x_n]$ be a $n$-dimensional vector positioned at a grid point. We define the *elementary effect* on the $i^{\mathrm{th}}$ variable as

$$E_i = \frac{f(\mathbf{x} + \Delta \mathbf{e_i}) - f(x_i)}{\Delta}, \tag{3.3}$$

where $\mathbf{e_i}$ is the canonical unit vector in the $i^{\mathrm{th}}$ direction. In Morris' standard method the step size $\Delta$ is chosen $\Delta = g/(2(g-1))$ with $g > 3$, hence defining a global SA method. Morris' goal was to calculate as many elementary effects as possible with the least number of model evaluations. He realized that it is possible to calculate $n$ elementary effects from $n+1$ samples by creating a "trajectory" of length $n+1$ in the following way: One starts at a random grid point $\mathbf{x}^{(0)}$, chooses a random canonical direction $\mathbf{e}_i$, and moves with step size $\pm\Delta$ along the

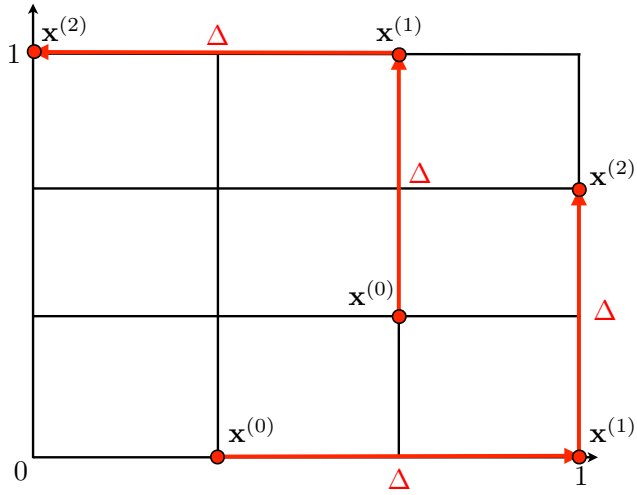



Figure 3.1: Two example trajectories of the Morris' method in 2D with $g = 4$.

direction. Two constraints have to be fulfilled: (i) Each canonical direction is only chosen once and (ii) if an attempted step leads to a point outside the domain, the reverse direction is chosen. If the new point $\mathbf{x}^{(1)}$ satisfies the constraints, it is used as the starting point for the next step. A 2D illustration of this process is given in Fig. 3.1. From a complete trajectory $(\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(j)}, \ldots, \mathbf{x}^{(n)})$, one elementary effect can be calculated for each variable:

$$E_i = \frac{f(\mathbf{x}^{(j)}) - f(\mathbf{x}^{(j-1)})}{\Delta}, \tag{3.4}$$

assuming the $i^{\text{th}}$ direction has been chosen in step $j$. Campolongo (Campolongo et al., 2004) has shown that defining the elementary effect as the absolute value of the function difference is more informative than the original definition, hence

$$E_i^* := \left| \frac{f(\mathbf{x}^{j}) - f(\mathbf{x}^{j-1})}{\Delta} \right|. \tag{3.5}$$

This is the definition we consider here. In order to avoid aliasing effects between the grid spacing and the frequencies present in the objective function, we further abandon the restriction of the starting point being located at a grid point. We rather choose a starting point uniformly at random in the landscape domain. $\Delta$ is chosen to be the limit $\lim_{g\to\infty} g/(2(g-1)) = 1/2$. This implies that the samples from all trajectories represent an unbiased, yet correlated uniform sample from the domain. This is an appealing property as both FDC and function dispersion can be calculated from these samples as well. The allowed FES budget dictates the number of

trajectories $T$ that can be calculated. For each variable we calculate means $\mu_i$ and standard deviations $\sigma_i$ of the resulting $T$ elementary effects $E_i^{*,t}$ according to:

$$\mu_i = \frac{1}{T}\sum_{t=1}^{T} E_i^{*,t}\,, \tag{3.6}$$

$$\sigma_i^2 = \frac{1}{T-1}\sum_{t=1}^{T}(E_i^{*,t}-\mu_i)^2\,. \tag{3.7}$$

The $\mu_i$ are used to identify the relative importances of the different variables. A larger $\mu_i$ indicates that a change in the $i^{\text{th}}$ variable *anywhere* in the landscape domain has, on average, a larger effect on the objective function variation. The $\sigma_i$ can be used to identify variable interactions. A large $\sigma_i$ implies that the effect of varying the $i^{\text{th}}$ variable heavily depends on the position in space, thus suggesting an interaction between different parameters. The quantity $\sigma_i$ can only be used to assess whether a variable interacts with *any* other variable. More refined information about which groups of variables are interacting cannot be obtained from $\sigma_i$. For general landscape analysis, it is convenient to define condensed quantities that are independent of the dimension and of the absolute scale of the fitness function. We therefore suggest using the normalized total importance variation $t_\mu$ and the normalized total interaction variation $t_\sigma$ as useful landscape descriptors. The quanitiy $t_\mu$ is defined as:

$$t_\mu = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{\mu_i-\bar{\mu}_i}{\bar{\mu}_i}\right)^2}\,, \tag{3.8}$$

with $\bar{\mu}_i$ being the average importance $\mu_i$. We define $t_\sigma$ as:

$$t_\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{\sigma_i-\bar{\sigma}_i}{\bar{\sigma}_i}\right)^2}\,, \tag{3.9}$$

with $\bar{\sigma}_i$ being the average interaction $\sigma_i$. A small value of $t_\mu$ indicates that all variables are about equally important. For optimization this suggests that globally there is no preferred search direction. A small value of $t_\sigma$ suggests that the problem is separable. Thus, for identifying optimal fitness values, $n$ one-dimensional optimization runs might be a successful strategy on such landscapes.

## 3.3 Landscape ruggedness

In order to assess the ruggedness of a real-valued black-box landscape, we follow Weinberger's strategy of the Random Walk autocorrelation function (Weinberger, 1990), which has been introduced in the context of combinatorial landscapes. The general idea is to quantify the fitness correlation between "neighboring" positions in the landscape. Consider the continuous unit hypercube as landscape domain $\mathcal{X}$. We suggest to explore the continuous landscape domain by a random walk with fixed step length. We start the random walk at some point

$\mathbf{x}^{(0)} \in \mathcal{X}$ and evaluate the fitness $f^{(0)} = f(\mathbf{x}^0)$. The next direction is chosen uniformly at random from the unit hypersphere, and a step of fixed length $s$ is performed in this direction. The new sample point $\mathbf{x}^{(1)} \in \mathcal{X}$, $f^{(1)} = f(\mathbf{x}^1)$ is then added to the random walk trajectory. If $\mathbf{x}^{(1)}$ falls outside the landscape domain, a new random directions is chosen until $\mathbf{x}^{(1)} \in \mathcal{X}$. The random walk is continued until the FES budget $S$ is exhausted. Based on the continuous random walk trajectory $(\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(j)}, \ldots, \mathbf{x}^{(S-1)})$ and the corresponding fitness values $(f^{(0)}, \ldots, f^{(j)}, \ldots, f^{(S-1)})$ the autocorrelation function can be computed as

$$\rho_{\mathrm{RW}}(k) = \frac{E[f(x^{(j)})f(x^{(j+k)})] - E[f(x^{(j)}]E[f(x^{(j+k)})]}{Var(f(x^{(j)}))}\,, \tag{3.10}$$

for any lag $k < S - 1$. $E[\cdot]$ and $Var[\cdot]$ are estimators for the sample mean and the sample variance, respectively. The correlation length $\tau = -\frac{1}{\log \rho_{\mathrm{RW}}(1)}$ is used as a condensed statistical fingerprint of the landscape. A large $\tau$ implies long-distance correlations between fitness values, thus suggesting a smooth landscape. Conversely, a low value indicates a highly rugged landscape where little correlation is present between neighboring samples.

A crucial choice in the present method is the step length $s$. This length defines the neighborhood $\mathcal{N}(\mathbf{x}^{(j)})$ of the samples. In combinatorial optimization, the standard choice is a Hamming distance of 1. In continuous spaces, it is not clear how to choose $s$. Fortunately, we can use a result from computational geometry for setting $s$. A fundamental problem in computational geometry is the estimation of the volume of a high-dimensional convex body $K$ that is given by a membership oracle. This oracle (or black box) returns "yes" for a given sample $\mathbf{x}^{(j)}$ iff $\mathbf{x}^{(j)} \in K$ and "no" otherwise. In order to estimate the volume of $K$, randomized algorithms are used to generate samples from $K$. An efficient way (albeit not the most efficient way) to sample from $K$ is given by the *lazy ball walk*. This walk is identical to the random walk presented above except that the steps are not selected from the surface of the sphere or radius $s$ but from entire volume of the ball with radius $s$ (Lovász, 1999). In order to sample the *entire* $K$ as fast and efficiently as possible, the optimal ball radius $s$ is dimension-dependent and must satisfy $s < \frac{1}{\sqrt{n}}$ (Lovász, 1999). We thus suggest to use the standard setting $s = \frac{1}{2\sqrt{n}}$ for the fixed step size random walk.

Note that the points generated by this random walk represent an unbiased uniform sample from the landscape domain that can also be used to derive dispersion and FDC information. Also, this landscape estimator is not restricted to box-constrained problems. The random walk method equally works for arbitrary *convex* landscape domains.

## 3.4 Characterization of the CEC 2005 benchmark test suite

We test the presented statistical landscape descriptors on all functions of the CEC 2005 benchmark test suite except $f_7$ and $f_{25}$. The latter problems are unconstrained and our statistical characterization framework is hence not applicable unless some user-specific region of interest is defined. However, note that apart from the missing constraints $f_{25}$ is identical to $f_{24}$. We choose the CEC 2005 benchmark because (i) the global topology, separability, and ruggedness

properties of most functions are known *a priori*, and (ii) it allows researchers to relate algorithmic performance to the calculated landscape descriptors. We thus consider $n = 10, 30, 50$ with the standard restriction on the FES budget (MAX_FES $= 10^4 n$) and 25 repetitions per run.

**Fitness-distance correlation.** We first present scatter plots of the fitness and distance data for all considered CEC functions in Fig. 3.2. We focus on the 10-dimensional case. The scatter plots look similar also in higher dimensions (data not shown). Visual inspection of the plots in Fig. 3.2 reveals a rich diversity of patterns. Function $f_1$ can be clearly identified as the sphere function. Fitness-distance plots of $f_6$, $f_9$, and $f_{10}$ show strong positive correlations. For functions $f_8$, $f_{11}$, and $f_{14}$ the spherical scatter patterns suggest a complete absence of correlation. Functions $f_2$–$f_5$, $f_{12}$, and $f_{13}$ show a similar pattern, suggesting weak correlations

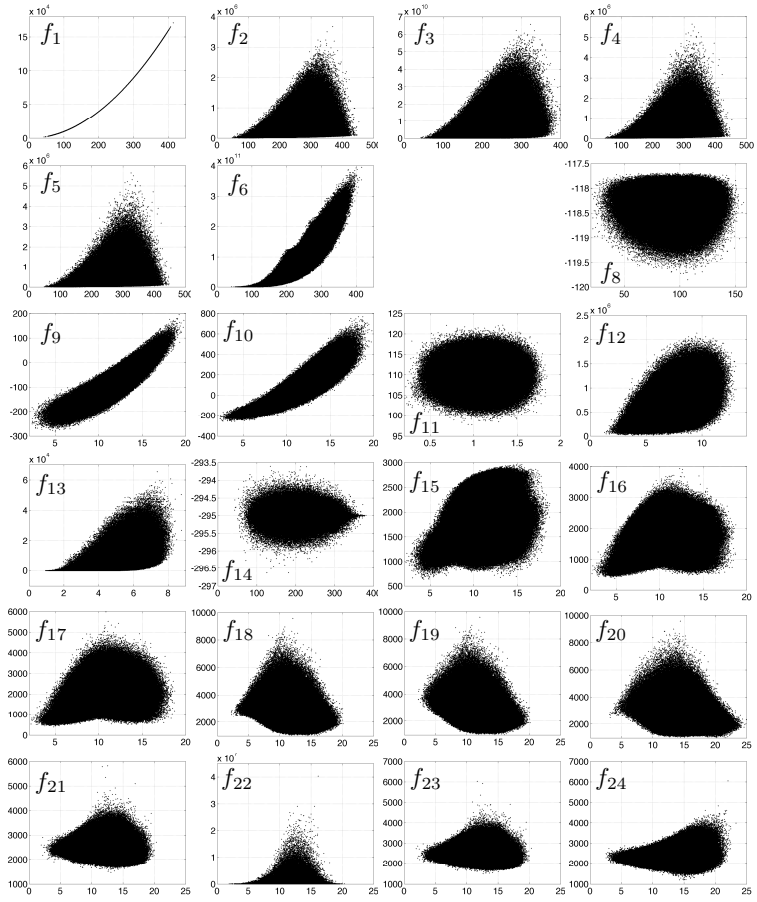



Figure 3.2: Fitness $f_i$ versus distance to the global minimum $d_{\mathrm{E}}(\mathbf{x}_{\min}, \mathbf{x})$ for all CEC functions except $f_7$ and $f_{25}$ in $n = 10$ dimensions. The FES budget is limited to $10^4 n$. The pooled samples from all 25 repetitions are shown.

between fitness and distance in all these cases. The scatter plots for $f_{18}$–$f_{24}$ reveal that many samples far away from the minimum have considerably lower objective function values than samples close to the global minimum, characterizing these problems as "deceiving". An unique scatter plot pattern is observed for the triplet $f_{15}$–$f_{17}$. For samples with low objective function values, two distinct distance regimes are visible, which may suggest a double-funnel topology of the landscape.
We summarize the calculated FDC coefficients $r_{\mathrm{FD}}$ in Fig. 3.3. The data suggest a rough

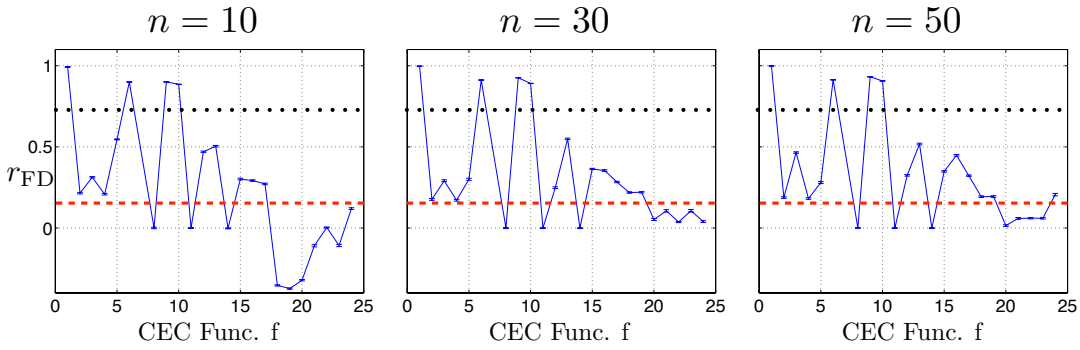



Figure 3.3: Estimated FDC coefficients (mean and std) for all CEC functions except $f_7$ and $f_{25}$ in $n = 10, 30, 50$ dimensions. The FES budget is limited to $10^4 n$. The black dotted line and the red dashed line represent the classification thresholds (see main text).

classification of the functions into three classes: (i) highly correlated $r_{\mathrm{FD}} > 0.75$, (ii) weakly correlated $0.75 > r_{\mathrm{FD}} > 0.15$, and (iii) uncorrelated or anti-correlated $r_{\mathrm{FD}} < 0.15$ across all dimensions. Only functions $f_{18}, f_{19}$, and $f_{24}$ change class in higher dimensions.

The functions $f_1$, $f_6$ and $f_9 - f_{10}$ belong to the first class. This suggests a global single-funnel topology. The shifted sphere function $f_1$ is expected to follow this classification. The shifted/rotated Rosenbrock function $f_6$, however, is multimodal. Nonetheless, the $r_{\mathrm{FD}}$ suggests that this multi-modality only appears at small length scales. The Rastrigin pair $f_9/f_{10}$ is also expected to give large $r_{\mathrm{FD}}$ values because of its globally spherical structure. Comparing the two functions of this pair also reveals that the rotation in $f_{10}$ does not significantly change the estimated $r_{\mathrm{FD}}$ values.

In all dimensions, the set of weakly correlated functions comprises functions $f_2$–$f_5$, $f_{12}$–$f_{13}$, and $f_{15}$–$f_{17}$. While $f_2$–$f_5$ are unimodal functions, the others are highly multimodal with little or no globally convex structure. $r_{\mathrm{FD}}$ values cannot discriminate these functions. The similar $r_{\mathrm{FD}}$ values for function pair $f_2/f_4$ and $f_{16}/f_{17}$ indicate that the measure is robust against noise. Among all hybrid functions ($f_{14}$–$f_{25}$), the $r_{\mathrm{FD}}$ suggest that the triplet $f_{15}$–$f_{17}$ has the highest degree of global correlation.

The class of un-/anti-correlated contains $f_8$, $f_{11}$, $f_{14}$, and $f_{20}$–$f_{23}$ across all dimensions. For these functions, a low fitness-distance correlation is expected. For instance, $f_8$ is a needle

problem and $f_{14}$'s global minimum is surrounded by regions of alternating high and low objective function values whose amplitude decreases with increasing distance (see Fig. 2.16). The $r_{\mathrm{FD}}$ values for the pair $f_{18}/f_{19}$ change from anti-correlation in $n = 10$ to weak correlation in $n = 30, 50$. This indicates that certain topological features that have been picked up by the measure in $n = 10$ dimensions cannot be detected any more in higher dimensions.

**Dispersion moments.** We present the results for the mean dispersion difference $\Delta^{\mathrm{m}}_{\mathrm{dis}}(f) = dis^{\mathrm{m}}(100, 10^4 n, f) - dis^{\mathrm{m}}(100, 100, f)$ in Fig. 3.4 and for the variance dispersion difference $\Delta^{\mathrm{v}}_{\mathrm{dis}}(f) = dis^{\mathrm{v}}(100, 10^4 n, f) - dis^{\mathrm{v}}(100, 100, f)$ in Fig. 3.5.

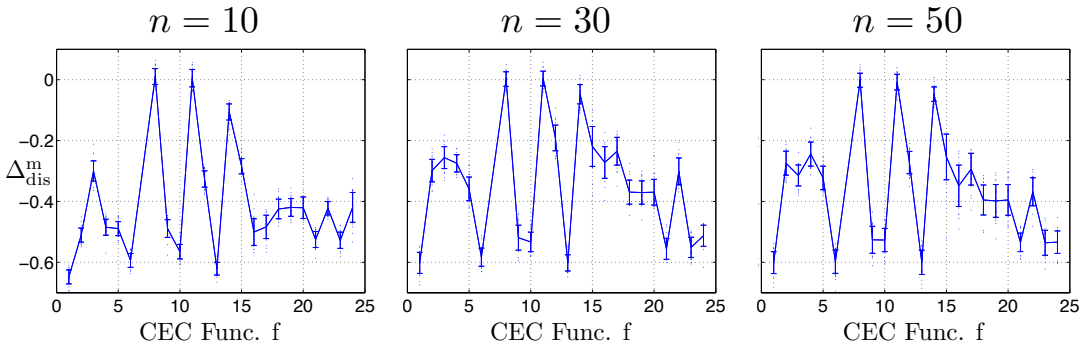



Figure 3.4: Estimated $\Delta^{\mathrm{m}}_{\mathrm{dis}}(f)$ (mean and std) for all CEC functions except $f_7$ and $f_{25}$ in $n = 10, 30, 50$ dimensions. The FES budget is limited to $10^4 n$.

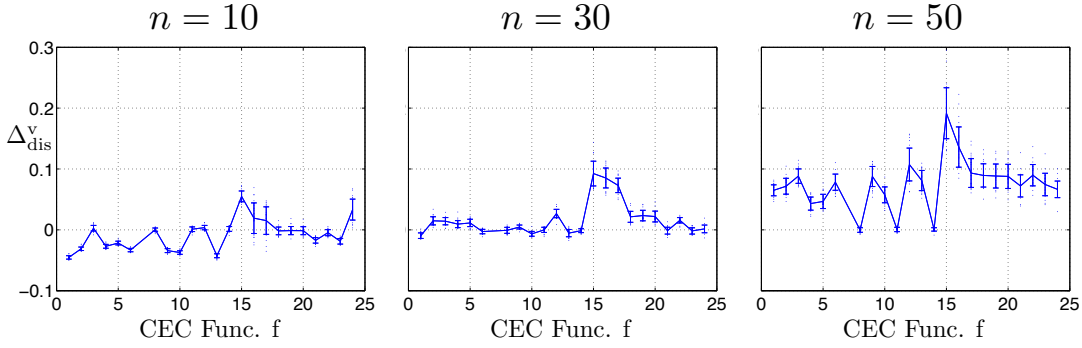



Figure 3.5: Estimated $\Delta^{\mathrm{v}}_{\mathrm{dis}}(f)$ (mean and std) for all CEC functions except $f_7$ and $f_{25}$ in $n = 10, 30, 50$ dimensions. The FES budget is limited to $10^4 n$.

The dispersion results mostly confirm the previous analysis using FDC. Across all dimensions, the functions $f_1$ $f_6$, $f_9$, $f_{10}$, and $f_{13}$ have a mean dispersion $< -0.5$, suggesting a global single-funnel topology. Likewise, the functions $f_8$, $f_{11}$, and $f_{14}$ being highly dispersive agree with their observed low FDC. The smooth unimodal functions $f_2$–$f_5$ have a dispersion pattern similar to the hybrid functions, suggesting that the mean dispersion difference alone cannot discriminate

between these very different topologies. In combination with the variance dispersion difference $\Delta^{\mathrm{v}}_{\mathrm{dis}}(f)$, however, a notable difference for the triplet $f_{15}$–$f_{17}$ is observed.These functions have a double-funnel topology. It is, hence, expected that the set of all pairwise distances between selected samples both contain very small and very large distances. The variance at the lowest threshold should thus be higher than the initial variance. This signal is picked up by $\Delta^{\mathrm{v}}_{\mathrm{dis}}(f)$, most prominently in $n = 30$ (see middle panel in Fig. 3.5.) Like FDC, the mean dispersion difference is robust against noise and rotation of the landscape domain.

**Morris' method.** The summary statistics of the two Morris' based landscape descriptors $t_\mu$ and $t_\sigma$ are presented in Fig. 3.6. As expected, the resulting pattern is different from the previous landscape descriptors. For all dimensions, $f_6$ has the largest value for $t_\mu$ followed by $f_{13}$ and $f_{22}$. No difference is observed between the class of unimodal, multi-modal, and hybrid functions. Noise lowers the estimated $t_\mu$ considerably, as reflected by the comparisons of the $f_2/f_4$ and $f_{16}/f_{17}$ values. The indicator for separability $t_\sigma$ can detect the separable functions $f_1$ and $f_9$ in all dimensions (see Fig. 3.7 for the $\sigma_i$ spectrum of $f_9$). In addition, low $t_\sigma$ values are also observed for $f_{14}$ and $f_{17}$. High $t_\sigma$ values are observed for $f_6$, $f_{13}$, and $f_{22}$. The function $f_{22}$ is a rotated version of $f_{21}$. The corresponding rotation matrix has a

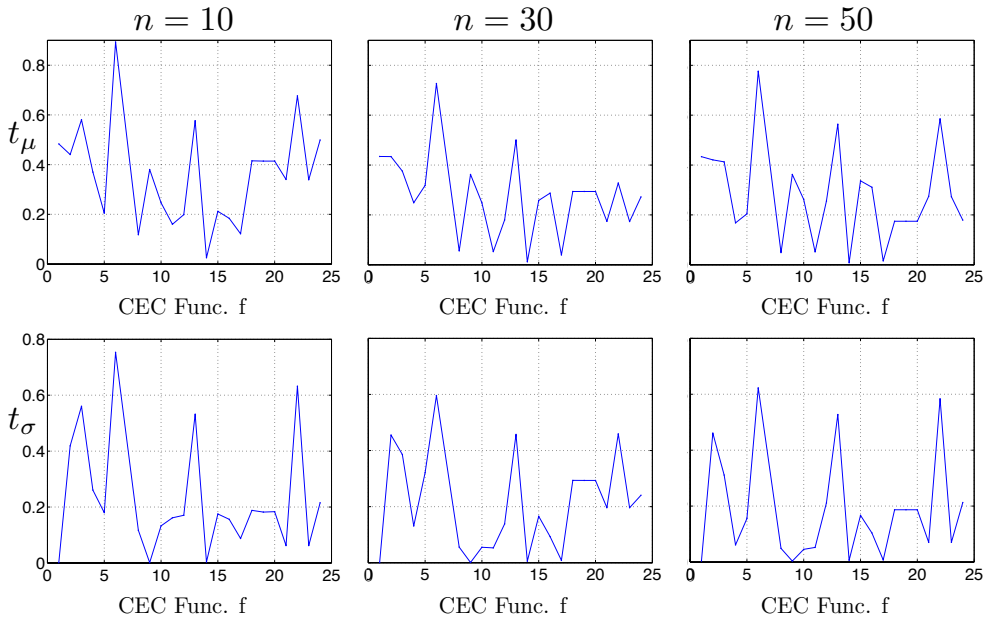



Figure 3.6: Estimated mean $t_\mu$ and $t_\sigma$ values for all CEC functions except $f_7$ and $f_{25}$ in $n = 10, 30, 50$ dimensions. The FES budget is limited to $10^4 n$.

high condition number, thus increasing variable interactions. Schwefel's double-sum function $f_2$ (see Fig. 2.15) also shows high $t_\sigma$ values across all dimensions. Its spectrum of Morris' interaction variables $\sigma_i$ is depicted in Fig. 3.7 for $n = 10$. $\sigma_i$ decreases with increasing index $i$, relating to the properties of the quadratic form that defines $f_2$ (Suganthan et al., 2005).

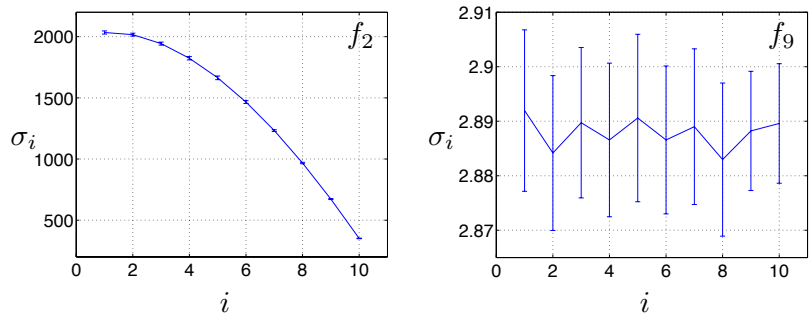



Figure 3.7: Variable interactions $\sigma_i$ for all variables $i$ of function $f_2$ and $f_9$ in $n = 10$ dimensions. $f_2$ shows decreasing interactions with increasing variable index. Function $f_9$ is separable.

**Random walk autocorrelation.** Landscape ruggedness is probed using the objective function autocorrelation of a random walk. The estimated correlation length $\tau$ serves as an indicator for the smoothness of the landscape. The smaller $\tau$ the more rugged the landscape. The computed autocorrelation coefficients $\rho_{\mathrm{RW}}(1)$ and the $\tau$ values are summarized in Fig. 3.8 for $n = 10$. In higher dimensions the observed pattern is similar. The quadratic functions $f_1$–$f_3$ have the largest measured correlation length. Noise reduces the correlation length (drop for $f_4$ and $f_{17}$). The needle problem $f_8$, the fractal Weierstrass function $f_{11}$, and $f_{14}$ show the smallest correlation lengths. For functions $f_{10}$ and $f_{16}$, the applied rotation *increases* the

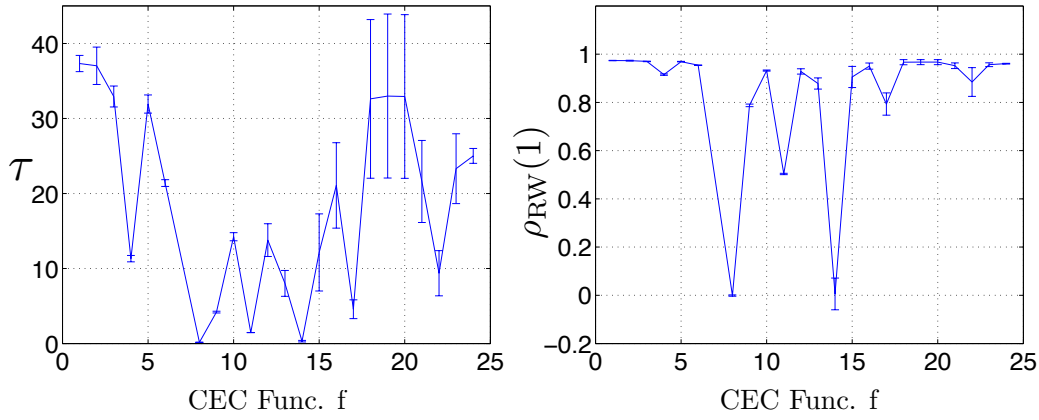



Figure 3.8: Correlation length $\tau$ and autocorrelation coefficient $\rho_{\mathrm{RW}}(1)$ (means and standard deviations) for all CEC functions in $n = 10$.

correlation length compared to $f_9$ and $f_{15}$. At least in the single-funnel case ($f_{10}$) this suggests that finding the global minimum is easier in the rotated problem than in the original one. The triplet $f_{18}$–$f_{20}$ shows a high average correlation length, but also a large standard deviation. This suggests that these landscapes are composed of distinct regions with high and low fitness autocorrelation.

## 3.5 Conclusions

We have introduced a set of statistical measures that allow characterizing continuous black-box landscapes. The following aspects of landscapes can be quantified: global landscape topology, variable importance, separability, and landscape ruggedness. Fitness-distance correlation and dispersion differences have been employed to study the global landscape topology. The Morris method, an OAT screening scheme that belongs to the class of global sensitivity analysis algorithms, has been introduced and enhanced in the present context. This allows probing variable importance and separability. We derived two dimension-independent scalar quantities that allow comparison between different landscapes: the normalized total importance variation $t_\mu$ and the normalized total interaction variation $t_\sigma$. Finally, a measure of ruggedness based on random walk autocorrelation has been introduced for continuous landscapes. This involves the correlation length of objective function values gathered by a specific geometric random walk.

We have applied all landscape descriptors to the complete function set of the IEEE CEC benchmark test suite in $n = 10, 30$, and 50 dimensions. Our results have shown that the CEC 2005 benchmark functions cover a wide spectrum of FDC coefficient values. This contradicts recent results of Vanneschi and co-workers (Vanneschi et al., 2010) who claim that the CEC 2005 test functions only have FDC coefficients close to one or zero. Fitness-distance correlation and dispersion differences can discriminate between functions with global single-funnel topology, such as the Rastrigin function, and highly unstructured problems like the needle problem $f_8$. However, the highly anisotropic ellipsoidal function $f_3$, although smooth and unimodal, cannot be discriminated from multi-funnel problems. Lunacek and Whitley argued that function dispersion is a predictive measure for the performance of CMA-ES (Lunacek and Whitley, 2006). CMA-ES can, however, efficiently solve the function $f_3$ (see Section 4.2.3). Our results thus challenge the general validity of Lunacek and Whitley's conclusions. The other landscape descriptors largely meet our expectations. The modified Morris method and its derived quantities can robustly identify separable functions across all dimensions, and the random walk autocorrelation coefficient captures the ruggedness of the landscapes.

We expect the presented landscape descriptors to be useful in a variety of situations. In model development and analysis, the Morris method can be used in the traditional sense for model reduction. Model parameters that show low $\mu_i$ and $\sigma_i$ can be fixed to appropriate constants. In optimization, an initial screen with a set of uniform samples in the landscape domain can be conducted, and the landscape descriptors can be applied in order to inform the modeler about the underlying problem landscape and provide guidance about (i) which optimizer to choose for the given problem and (ii) how to set internal strategy parameters of the optimizer. For instance, when the correlation length is low, robust stochastic optimizers may be preferred over schemes that rely on local information, such as approximate gradient descent schemes.

Finally, we argue that the presented statistical quantities can be used as landscape fingerprints or features in a classification scenario. We envision an unsupervised statistical learning framework that, given a list of samples and associated features, can infer a classification of black-box landscapes.

# 4

# Optimization of Black-box Landscapes

> Homer: "Kids, there's three ways to do things. The right way, the wrong way and the Max Power way."
> Bart: "Isn't that the wrong way?."
> Homer: "Yeah, but faster!"
> in: The Simpsons, Homer to the Max, Episode no. 216, 1999

## 4.1 Introduction

We now focus on the arguably most important landscape feature, the location of global optima. A large class of problems in science and engineering can be formulated as black-box optimization tasks. Typical instances include data fitting to nonlinear models and determination of design parameters in complex computer models. These problems often share the following characteristics: (i) Only zeroth-order information is available (a black-box or oracle). (ii) The budget of black-box evaluations is limited. (iii) The underlying global topology of the landscape is unknown. (iv) The landscape can be high-dimensional ($n \gg 10$), discontinuous, and corrupted by noise. Because the diversity of real-world problems prohibits a unifying mathematical classification, the development of search heuristics has been largely "driven by practical success whereas the aspect of theoretical analysis is neglected." (Jens Jägersküpper, 2008). This has led to an enormous variety of sophisticated and/or problem-specific algorithms with little or no theoretical justification. Nevertheless, there have been considerable attempts to identify and establish general design principles for efficient black-box optimization methods. We next sketch the most important concepts and relate them to our landscape perspective. After these introducing remarks, we present our main contributions to the field of black-box

optimization.

The development of black-box algorithms dates back to 1950's. Box and Wilson introduced the response surface methodology in their seminal paper "On the Experimental Attainment of Optimum Conditions" (Box and Wilson, 1951). Their goal was to optimize experimental conditions of chemical or biological processes. The controllable parameters are, for instance, temperature, pressure, time of reaction, and proportions of reactants. Their idea was to estimate a statistical model, the "response surface", in a local region of parameter space and use the estimated gradients for a steepest ascent step on the response surface. At the next location a new response surface is estimated. In a follow-up paper Box also suggested to consider evolutionary operations, such as variation and selection, in order to improve parameters in industrial processes (Box, 1957). Brooks was among the first to realize that *randomization* is key for a successful black-box method (Brooks, 1958). In his "Discussion of random methods for seeking maxima" he envisioned several optimization methods that are very close to modern heuristics. Hooke and Jeeves described the "pattern search" method (Hooke and Jeeves, 1961) where the term "direct search" method is introduced equivalently to our notion of a black-box algorithm. Nelder and Mead introduced the downhill simplex method (Nelder and Mead, 1965) where the shape of a simplex is continuously adapted in order to converge to a local minimum. Rastrigin (Rastrigin, 1963, 1972) introduced the *fixed step-size random search* (FSSRS) method. There, at each generation a single candidate solution that has a fixed distance to the previous one is sampled randomly and is accepted if its objective function value is better than the previous one. Schumer and Steiglitz realized that *adaptation* of the step size is fundamental for effective random search (Schumer and Steiglitz, 1968). Adaptation is coupled to the probability of accepting a new sample, $P_{\mathrm{acc}}$, where $P_{\mathrm{acc}} \approx 0.27$ is optimal under certain assumptions. Both fixed and *adaptive step size random search* (ASSRS) use uniformly distributed search directions. When a multivariate normal distribution is employed instead, ASSRS is equivalent to the well-known (1+1)-Evolution Strategy (ES) (Rechenberg, 1973; Schwefel, 1975), which started the field of Evolutionary Computation. From experiments on Rosenbrock's function, Schumer and Steiglitz commented on the influence of the landscape topography on the search performance. They concluded that "ASSRS is not very effective as a ridge follower, but shows its superiority in multidimensional problems without narrow valleys or ridges. Combining directional adaptation with step size adaptation may result in removing this limitation" (Schumer and Steiglitz, 1968). This suggestion is realized in *Variable-Metric Algorithms* that are ubiquitous in optimization.

### *Variable-metric approaches in optimization*

All variable metric approaches are iterative algorithms that share the idea of adapting a position vector and a quadratic form. At each step, the quadratic form defines a metric between gradients that reflect the *local structure* of the landscape. Davidon introduced in an Argonne National Laboratory Research and Development Report in 1959 the "Variable Metric Method" as *first-order method* for general non-linear, real-valued minimization (see (Davidon, 1991) for a commented reprint). The key idea is to "learn" the inverse of an appropriate Hessian matrix $H$ from analytic gradient vectors at *all* visited locations of the landscape domain.

The entries of the inverse Hessian encode both the current step size and the search direction, thus representing a continuously changing metric. In 1970, Broyden (Broyden, 1970), Fletcher (Fletcher, 1970), Goldfarb (Goldfarb, 1970), and Shanno (Shanno, 1970) independently derived an efficient update rule for the inverse Hessian that is now referred to as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. These types of methods are also called *Quasi-Newton methods* because they approximate the second-order Newton method where the full Hessian is explicitly calculated. The BFGS method and its limited-memory variant (Nocedal, 1980), where only a limited number of previous gradients is taken into account, are nowadays state-of-the-art in first-order real-valued function minimization. Variable metric approaches are also instrumental in the fields of convex and combinatorial optimization, as well as in linear programming (Goffin, 1984). The most prominent algorithm is the *Ellipsoid method* where an ellipsoid, described by a center vector and a quadratic form, is iteratively adapted. For real-valued convex minimization, the ellipsoid method generates a sequence of ellipsoids with changing orientation and position whose volume uniformly decreases at every step, while ensuring that the minimizer of the convex function is enclosed in the volume. A precursor of this method has been published by Shor (Shor, 1970). Yudin and Nemirovski (Yudin and Nemirovski, 1976) and Shor (Shor, 1976) studied the full ellipsoid method as an approximation algorithm. For linear programming, Khachiyan proved polynomial run time of the ellipsoid method (Khachiyan, 1979). We refer to (Grötschel et al., 1993; Papadimitriou and Steiglitz, 1998) for further details. To the best of our knowledge, the explicit use of the term "variable metric" in the context of black-box optimization has been introduced in (Suttorp et al., 2009) and the habilitation thesis of Hansen (Hansen, 2010b). Two classes of variable metric algorithms emerged in the black-box optimization field: (i) Methods that sequentially learn an explicit local quadratic approximation of the objective function, and (ii) stochastic methods that adapt the covariance matrix of a multivariate search distribution. Methods of the first kind are also called "trust region methods". Powell's "Unconstrained Optimization by Quadratic Approximation" (UOBYQA) method (Powell, 2002) and its successor NEWUOA (Powell, 2006) are important representatives. Kjellström's Gaussian Adaptation (*GaA*) algorithm (Kjellström and Taxen, 1981) and Hansen's Evolution Strategy with Covariance Matrix Adaptation (*CMA-ES*) (Hansen and Ostermeier, 2001; Hansen et al., 2003; Hansen, 2008) belong to the second class. These state-of-the-art black-box methods, along with our extensions and improvements, are the topic of Sections 4.2 and 4.4.

### *Design principles for black-box optimization*

The variable-metric approach is one prerequisite for an algorithm to yield certain *invariance* properties with respect to the underlying landscape structure. Achieving invariance properties constitutes a useful design principle for black-box search. Invariance has been proposed by Hansen (Hansen, 2000) in the ES context. In his habilitation thesis (Hansen, 2010b) Hansen states: "In search, invariance properties induce equivalence classes of objective functions, on which the performance of the search algorithm is identical. Consequently, any result observed on a real world problem, or on a test function, does not only hold for this single problem instance, but inevitably generalizes to the complete class of problems induced by the invariance property, thereof the tested problem is an element. Hence stronger statements on the

performance of the search algorithm can be made – a greater number of empirical facts is covered. The drawback to invariance properties in search is that whenever an invariance property is achieved, some information cannot be exploited anymore." Two kinds of invariance properties are common: *invariance under function value transformations* and *invariance under search space transformations*. We revisit the invariance properties of CMA-ES and GaA in Sections 4.2 and 4.4. Further definitions and examples can be found in (Hansen, 2000) and (Hansen, 2010b). Our landscape perspective also suggests an additional design principle: robust performance on problems with a multi-funnel structure. It is conceivable that such insensitivity is harder to achieve than the other design principles. In fact, complete independence of search performance and landscape topology can only be achieved by pure random search. This is also the essence of the No-Free-Lunch theorem (Wolpert and Macready, 1997). We argue that strategies that explore the landscape in parallel and in a collaborative manner can relax the negative effect of multi-funnel structure on search performance. This is exploited in the parallel CMA-ES schemes developed in Section 4.3

## 4.2 The Covariance Matrix Adaptation Evolution Strategy

Since their first formulation in the 1960's by Rechenberg and Schwefel (Rechenberg, 1973), *Evolution Strategies* (ES) have been among the best-studied black-box optimization paradigms for non-convex, real-valued functions. A particularly successful instance is the Evolution Strategy with Covariance Matrix Adaptation (CMA-ES ). It has proven its usefulness in hundreds of real-world applications (Hansen, 2009b) and is considered the state of the art in black-box optimization. We revisit key algorithmic mechanisms of CMA-ES and present our enhancements to the method. We also present extensive numerical benchmark results that have been obtained with pCMALib, our novel parallel Fortran90 library that implements a family of CMA-ES methods. Technical details about this library are summarized in Appendix A2.

### 4.2.1 Canonical CMA-ES

Since CMA-ES is a Evolution Strategy, it is straightforward to view the search process as the evolution of a population on a fitness landscape. The population consists of individuals (samples, candidate solutions) that possess a phenotypic fitness arising from the underlying landscape. The evolutionary principles of *selection*, *mutation*, and *recombination* act on the population, resulting in a change of its location and overall "shape" from generation to generation. The population is eventually climbing up a nearby fitness mountain and converges to its peak. Figure 4.1 sketches this process on a unimodal landscape. Mathematically, the evolutionary search process is modeled as follows: The population is represented by a multivariate normal distribution $\mathcal{N}$ with mean $\mathbf{m} \in \mathbb{R}^n$ and covariance $\mathbf{C} \in \mathbb{R}^{n \times n}$. At each iteration (generation) of the algorithm, the members of a new population are sampled from this distribution. The normal distribution is appealing because (i) it is completely specified by its first two moments and (ii) Jaynes' maximum entropy principle (Jaynes, 1957) implies that, for given fixed mean and covariance, the normal distribution is the least biased choice among all distributions with the same mean and covariance. The latter concept is thoroughly discussed in Section 4.4. The number of samples, i.e., the population size $\lambda$, is constant over time. The

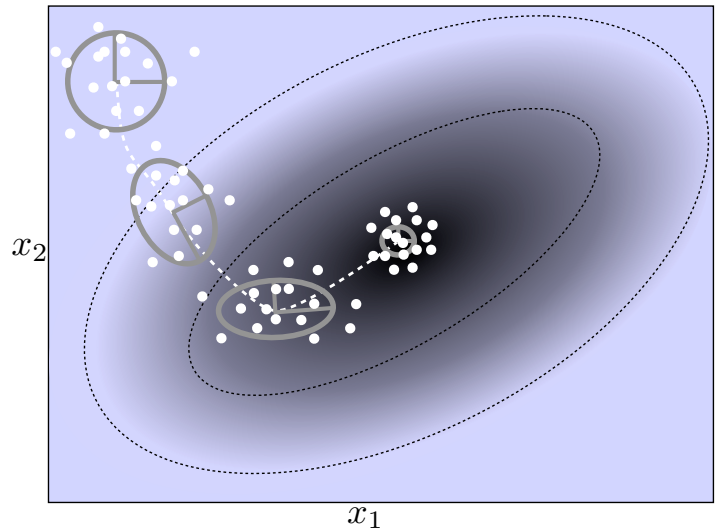



Figure 4.1: Illustration of the evolution of a CMA-ES population on a unimodal fitness landscape. The white dots represent the population members, the gray ellipsoids the population covariances. The white dashed line is the evolution path.

sampling radius is controlled by the overall standard deviation (step size) $\sigma$. Let $\mathbf{x}_k^{(g)}$ be the $k^{\text{th}}$ individual at generation $g$. The new individuals at generation $g+1$ are sampled as:

$$\mathbf{x}_k^{(g+1)} \sim \mathbf{m}^{(g)} + \sigma^{(g)} \mathcal{N}\left(\mathbf{0}, \mathbf{C}^{(g)}\right) \quad k = 1, \ldots, \lambda\,. \tag{4.1}$$

Selection is realized by ranking the $\lambda$ sample points in order of ascending fitness, and choosing the $\mu$ best individuals. This procedure is called *truncation selection*. The means of the sampling distribution is updated using *weighted intermediate recombination* of these selected candidates:

$$\mathbf{m}^{(g+1)} = \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{(g+1)} \tag{4.2}$$

with

$$\sum_{i=1}^{\mu} w_i = 1, \quad w_1 \geq w_2 \geq \ldots \geq w_\mu > 0\,, \tag{4.3}$$

where the $w_i$ are positive weights and $\mathbf{x}_{i:\lambda}^{(g+1)}$ denotes the $i^{\text{th}}$-ranked individual of the $\lambda$ sampling points $\mathbf{x}_k^{(g+1)}$. Super-linearly decreasing weights with $w_i = \log(\frac{\lambda-1}{2} + 1) - \log(i)$ are the standard choice. An ES with this selection scheme is termed $(\mu_\text{w}/\mu, \lambda)$-ES. Because selection only depends on the relative *ranking* of individuals within a population, CMA-ES is invariant to any strictly monotonic (that is, order-preserving) transformation of the objective function. It is important to note that such a scheme does not *per se* ensure that the mean fitness decreases in every generation. In order for CMA-ES to search efficiently, local correlation between fitness and sample locations must be present.

### *Covariance Matrix Update*

A fundamental ingredient of CMA-ES is its mechanism to learn meaningful search directions. This is realized by adapting the covariance matrix of the sample distribution. The goal of the covariance adaptation is to increase the likelihood of reproducing previously successful steps. Three sub-procedures are dedicated to covariance matrix adaptation: a *rank-$\mu$ update*, a *rank-one update*, and *cumulation*.

**Rank-$\mu$ update.** The rank-$\mu$ update of the covariance matrix has been introduced in (Hansen et al., 2003). One possibility for such an update is to use the empirical covariance matrix of the selected samples:

$$\mathbf{C}_{\text{emp}}^{(g+1)} = \frac{1}{\mu - 1} \sum_{i=1}^{\mu} \left( \mathbf{x}_i^{(g+1)} - \frac{1}{\mu} \sum_{j=1}^{\mu} \mathbf{x}_j^{(g+1)} \right) \left( \mathbf{x}_i^{(g+1)} - \frac{1}{\mu} \sum_{j=1}^{\mu} \mathbf{x}_j^{(g+1)} \right)^T. \tag{4.4}$$

However, such an update comprises information about successful *positions* rather than directions. The idea in CMA-ES hence is to consider the true mean of the sample distribution rather than the empirical mean of the selected samples. Including also the corresponding weights leads to the rank-$\mu$ update used in CMA-ES:

$$\mathbf{C}_{\mu}^{(g+1)} = \sum_{i=1}^{\mu} w_i \left( \mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{m}^{(g)} \right) \left( \mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{m}^{(g)} \right)^T \tag{4.5}$$

This update can be interpreted as a weighted estimator for the distribution of selected steps. This information is combined with the previous covariance matrix by introducing the learning rate $c_{\text{cov}}$:

$$\begin{aligned} \mathbf{C}^{(g+1)} &= (1 - c_{\text{cov}}) \mathbf{C}^{(g)} + c_{\text{cov}} \frac{1}{\sigma^{(g)^2}} \mathbf{C}_{\mu}^{(g+1)} \\ &= (1 - c_{\text{cov}}) \mathbf{C}^{(g)} + c_{\text{cov}} \sum_{i=1}^{\mu} w_i \mathbf{y}_{i:\lambda}^{(g+1)} \mathbf{y}_{i:\lambda}^{(g+1)^T}, \end{aligned} \tag{4.6}$$

where $\mathbf{y}_{i:\lambda}^{(g+1)} = (\mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{m}^{(g)})/\sigma^{(g)}$.

**Rank-One update and cumulation.** Rank-one update has been introduced in (Hansen and Ostermeier, 2001). It uses only a single selected step for covariance matrix adaptation. In order to exploit correlations between consecutive steps *the evolution path* of the search process is considered. The evolution path is the trace that the strategy has taken over a number of generations. It can be expressed as a sum over consecutive steps of the mean value $\mathbf{m}^{(g)}$ (Hansen, 2008). Recursive construction of the evolution path $\mathbf{p}_c^{(g+1)} \in \mathbb{R}^n$ with $\mathbf{p}_{\text{c}}^{(0)} = \mathbf{0}$, is referred to as *cumulation*:

$$\mathbf{p}_{\text{c}}^{(g+1)} = (1 - c_{\text{c}}) \mathbf{p}_{\text{c}}^{(g)} + \sqrt{c_{\text{c}}(2 - c_{\text{c}}) \mu_{\text{eff}}} \, \frac{\mathbf{m}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}}, \tag{4.7}$$

where $c_c \leq 1$ is the backward time horizon, and $\mu_{\text{eff}}$ is a measure paraphrased as variance-effective selection mass:

$$\mu_{\text{eff}} = \left( \sum_{i=1}^{\mu} w_i^2 \right)^{-1} . \tag{4.8}$$

**Combined adaptation.** Combining rank-$\mu$ update and rank-one update with cumulation results in the final covariance matrix update rule:

$$\begin{aligned} \mathbf{C}^{(g+1)} = (1 - c_{\text{cov}})\mathbf{C}^{(g)} + \frac{c_{\text{cov}}}{\mu_{\text{cov}}} \underbrace{\mathbf{p}_c^{(g+1)} \mathbf{p}_c^{(g+1)^T}}_{\text{rank-one-update}} + c_{\text{cov}} \left(1 - \frac{1}{\mu_{\text{cov}}}\right) \\ \times \underbrace{\sum_{i=1}^{\mu} w_i \mathbf{y}_{i:\lambda}^{(g+1)} \left(\mathbf{y}_{i:\lambda}^{(g+1)}\right)^T}_{\text{rank-}\mu\text{-update}} , \end{aligned} \tag{4.9}$$

with $\mu_{\text{cov}} \geq 1$ denoting the relative weight of the two updates.

### *Step-Size Control*

As suggested in ASSRS and the (1+1)-ES, the idea of step size adaptation is also adopted in CMA-ES. The information contained in the evolution path is used to control the overall scale $\sigma^{(g)}$ of the sampling distribution. A long evolution path implies that recent consecutive steps are pointing in a similar direction, i. e., they are correlated. In these cases CMA-ES increases $\sigma^{(g)}$. Likewise, a short evolution path implies consecutively anti-correlated steps, suggesting a decrease of the step-size. Similar to Eq. (4.7), a recursive formula for the step-size evolution path $\mathbf{p}_\sigma$ can be constructed:

$$\mathbf{p}_\sigma^{(g+1)} = (1 - c_\sigma)\mathbf{p}_\sigma^{(g)} + \sqrt{c_\sigma(2 - c_\sigma)\mu_{\text{eff}}}\ \mathbf{C}^{(g)^{-\frac{1}{2}}}\ \frac{\mathbf{m}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}} , \tag{4.10}$$

with $\mathbf{p}_\sigma^{(0)} = \mathbf{0}$. The backward time horizon of the evolution path is $c_\sigma < 1$. When $c_\sigma = 1$, only the most recent step contributes to the cumulation. The difference $\mathbf{m}^{(g+1)} - \mathbf{m}^{(g)}$ gives the current step, and $\sqrt{c_\sigma(2 - c_\sigma)\mu_{\text{eff}}}/\sigma^{(g)}$ is a normalization constant. The transformation $\mathbf{C}^{(g)^{-\frac{1}{2}}}$ can be found by eigendecomposition of the covariance matrix. This renders the expected length of $\mathbf{p}_\sigma^{(g+1)}$ independent of the current direction. The step-size adaptation rule is derived from the idea that the optimal length of the evolution path, $||\mathbf{p}_\sigma^{(g+1)}||$, is equal to its expected length under random selection. This length is, in fact, the expected length of a standard normal random vector $E||\mathcal{N}(\mathbf{0}, \mathbf{I})||$. The step size adaptation mechanism, hence, reads:

$$\sigma^{(g+1)} = \sigma^{(g)} \exp\left( \frac{c_\sigma}{d_\sigma} \left( \frac{||\mathbf{p}_\sigma^{(g+1)}||}{E||\mathcal{N}(\mathbf{0}, \mathbf{I})||} - 1 \right) \right) , \tag{4.11}$$

where $d_\sigma$ is a damping factor.

Step size and covariance matrix adaptation render CMA-ES approximately invariant to any linear transformations of the landscape domain. For instance, if the objective function is an ellipsoid with condition number $> 1$, CMA-ES is able to learn the eigenvectors of the underlying quadratic topology. This implies that search progress on any ellipsoid is, after a burn-in phase, identical to the progress on the sphere function. From the variable metric perspective, CMA-ES can also be interpreted as a method that learns the optimal Mahalanobis distance (Eq. (2.2)) for the search directions, given the current mean, step size, and covariance matrix.

### *Parameter Settings, Modes of Operations and Boundary Handling*

CMA-ES has several internal strategy parameters. Over the past decade, Hansen and co-workers derived standard parameter settings based on both theoretical arguments on quadratic functions and extensive numerical experimentation on more complex model landscapes, such as Rastrigin and Rosenbrock function (see (Hansen, 2008) and (Hansen, 2010b) for a complete list of parameter settings). In practice, the only remaining parameters that are of interest to the user are the population size $\lambda$ and the initial step size $\sigma^{(0)}$. The default population size is $\lambda = 4 + \lfloor 3 \ln n \rfloor$, where $n$ is the dimension of the problem. For multi-modal and noisy landscapes, CMA-ES also requires parametric rules that define convergence or divergence of the search trajectory. This is realized by storing the history of previous positions, fitness values, and condition numbers of the covariance matrix, and providing tolerances for these values (Hansen, 2008). When any of the tolerance criteria are met and the available FES budget is not exhausted, CMA-ES is restarted. Together with the free parameters $\lambda$ and $\sigma^{(0)}$ different modes of operation can then be defined. Auger and Hansen suggest two strategies: CMA-ES with iteratively increasing population size (IPOP-CMA-ES) (Auger and Hansen, 2005b) and Local Restart CMA-ES (LR-CMA-ES) (Auger and Hansen, 2005a). In IPOP-CMA-ES, the default population size $\lambda$ is doubled at each restart. In box-constrained optimization, the initial $\sigma^{(0)}$ is set to 50% of the largest box length. LR-CMA-ES always uses the default population size and a small $\sigma^{(0)}$ (0.5% of the largest box length) in order to perform local search. Both algorithms have been tested on the IEEE CEC 2005 benchmark contest where IPOP-CMA-ES has shown the best performance among all tested algorithms. We revisit the IPOP-CMA-ES results in Section 4.2.3.

Another important ingredient in CMA-ES is boundary handling. Most black-box problems are formulated with box constraints. When sampling new candidate solutions from a multivariate normal distribution it is possible that samples are placed outside the feasible region. How to handle these infeasible solutions is a research topic in its own right. We refer to (Kramer, 2010) for a recent review in the ES context. Several constraint handling techniques for CMA-ES are available. The simplest approach is to discard out-of-bounds samples and re-sample until the candidate point lies within the feasible region. This can, however, become inefficient especially in high dimensions, since the probability of hitting the feasible region decreases with dimensionality. Other approaches thus rely on adding an artificial penalty to the original objective function. This penalty usually is a monotonically increasing function of the Euclidian distance between the sample point and the nearest boundary. In CMA-ES, a sophisticated mechanism

is available to estimate the optimal scale of the penalty from samples of the objective function (Hansen, 2008; Hansen et al., 2009). We use this boundary handling throughout the thesis.

### 4.2.2 Novel CMA-ES variants

We present three novel CMA-ES variants. The first one, Low-discrepancy CMA-ES, uses specific number sequences instead of pseudo-random numbers for the generation of normal variates. The second, Best Local Restart CMA-ES, considers an alternative restart strategy that is applicable in cases where the black-box landscape contains deep narrow funnels. The third strategy, memetic CMA-ES, hybridizes CMA-ES with gradient-based techniques.

#### *Low-discrepancy CMA-ES*

In every CMA-ES generation, the population of candidate solutions $\mathbf{x}_k^{(g+1)}$ is drawn from a multivariate normal distribution $\mathcal{N}\Big(\mathbf{m}^{(g)}, \sigma^{(g)2}\mathbf{C}^{(g)}\Big)$. The general numerical procedure to draw these samples is

$$\mathbf{x}_k^{(g+1)} \sim \mathbf{m}^{(g)} + \sigma^{(g)}\mathbf{A}\,\mathcal{N}\Big(\mathbf{0}, \mathbf{I}_n\Big) \quad k = 1, \ldots, \lambda\,, \tag{4.12}$$

where $\mathbf{A}$ is the Cholesky matrix, such that $\mathbf{C} = \mathbf{A}\mathbf{A}^T$. Samples from the $n$-dimensional standard multivariate normal distribution $\mathcal{N}\Big(\mathbf{0}, \mathbf{I}_n\Big)$ can be generated independently for each dimension. A standard normal variate $\mathcal{N}\,(0, 1)$ is numerically generated as the Box-Muller transformation (Box and Muller, 1958) of a standard uniform random number. In the computer, independent uniform random numbers are approximated by *pseudo-random number generators* (PRNG) such as the Mersenne Twister (Matsumoto and Nishimura, 1998). Such pseudo-random number sequences are deterministic by construction, but they are considered a sufficient proxy for true randomness. *Low-discrepancy sequences* share the property of deterministic generation, but they are designed to produce a stream of numbers that fill the space as evenly as possible, rather than appearing random. Figure 4.2 shows pseudo-random and low-discrepancy (LD) samples of size $1e4$ in the unit square. Conceptually, the "discrepancy" of a sequence measures how uniformly all partitions of the unit hypercube are filled. If some regions are densely covered and others sparsely with big gaps between neighboring samples, the sample is considered highly discrepant. Several different sequences are available, such as Halton sequences (Halton, 1960), Faure sequences (Faure, 1992) and Sobol' sequences (Sobol, 1967). LD sequences are also referred to as *Quasi-random numbers*. The "quasi" indicates that they are *not* random, but are often used as a replacement for random numbers, for instance, in the numerical approximation of high-dimensional integrals in physics (Morokoff and Caflisch, 1995; Schlier, 2004) and finance (Caflisch et al., 1997; Joy et al., 1996) and in the context of continuous optimization (Biester et al., 1995; Kucherenko and Sytsko, 2005; Drew and de Mello, 2006; Liu and Owen, 2006). Another important area of application is variance-based sensitivity analysis (Saltelli et al., 2000).

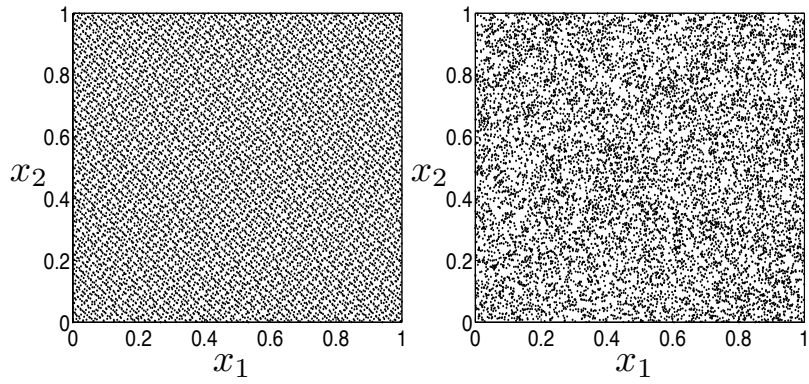



Figure 4.2: The first $10^4$ samples of MATLAB's Sobol sequences, and pseudo-random numbers generated by MATLAB's *rand* function in the unit square.

Although initially designed for uniformly covering the unit hypercube, it is possible to transform LD sequences into quasi-normal variates that retain the LD property. The Box-Muller transform is, however, not a good choice here.Rather Moro's inversion (Moro, 1995) or Acklam's inversion (Acklam, 2009) should be used. Quasi-normal samples show faster convergence to the true statistical moments and better covering of the tails of the Gaussian (Krykova, 2003). It is, thus, conceivable that a replacement of the standard pseudo-random sampling in CMA-ES by low-discrepancy sampling schemes may be beneficial for search performance. The working hypothesis for this low-discrepancy CMA-ES (LD-CMA-ES) is that, on average, the quasi-random samples cover the landscape domain more evenly, leading to faster *and* more robust convergence. The numerical routines are largely taken from (Burkardt, 2009) and implemented in pCMALib. Several combinations of samplers and transformations are implemented. By default we use Sobol sequences with Acklam's inversion. Numerical benchmark results are presented in Section 4.2.3. Note that Teytaud independently proposed the same idea in his DCMA (Derandomized CMA) scheme (Teytaud and Gelly, 2007; Teytaud, 2008), where specific types of Halton sequences are used.

### *Best Local Restart CMA-ES*

IPOP-CMA-ES and LR-CMA-ES are two restart variants that have been proven useful in practice. Common to both approaches is the idea of *independent* restarts. When any of the convergence criteria are met, CMA-ES is restarted totally oblivious to the information gathered in previous runs. The idea of such a setting is to consider the individual runs as random variables that have a certain probability of finding the global minimum. Independent restarts thus increase the probability of success in the general case. It is, however, conceivable that landscapes exist where such strategies fail. Consider a landscape topology that contains a number of deep, narrow funnels with rugged local structure and multiple local basins. Due to its small initial step size LR-CMA-ES is able to enter any of the funnels, but might prematurely

converge to a local basin before reaching the funnel bottom. IPOP-CMA-ES with increasing population will preferably enter and explore the widest funnel (Lunacek et al., 2008), which not necessarily contains the global minimum. Best Local Restart CMA-ES (BLR-CMA-ES) is designed to explore such landscapes efficiently. BLR-CMA-ES has three key components: First and foremost, local restarts are not conducted independently. At each restart, the starting point $\mathbf{x}^{(0)}$ is chosen to be the best candidate solution found so far rather than a random point. Only when several consecutive restarts converge to the same candidate solution, a random restart is done. Figure 4.3 sketches the possible search behavior of BLR-CMA-ES. The population size in BLR-CMA-ES is kept constant over restarts and scales linearly with the problem dimension. We suggest $\lambda_{\mathrm{BLR}} = 5\,n$. By default, a constant initial step size $\sigma^{(0)}_{\mathrm{BLR}}$ of 0.5% of the largest box length is used, representing the same setting as in LR-CMA-ES. If the correlation length $\tau$ of the landscape is known, $\sigma^{(0)}_{\mathrm{BLR}}$ may be set accordingly. These default

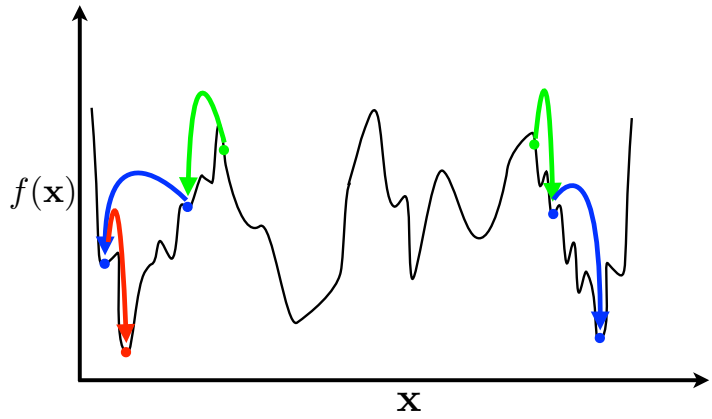



Figure 4.3: Sketch of a multi-funnel landscape. Two independent BLR-CMA-ES trajectories are depicted. The green dots mark the starting points of each BLR-CMA-ES instance. The blue dots indicate the best candidate solutions found before convergence. These candidate solutions serve as starting points for succeeding CMA-ES runs. The right BLR-CMA-ES run explores the sub-optimal funnel. The left BLR-CMA-ES run produces monotonically improving restart points until the bottom of the optimal funnel is reached (red dot).

settings have been derived from extensive numerical experiments on the black-box problems presented in Chapter 7. CEC 2005 benchmark results of BLR-CMA-ES with alternative parameter settings are available in (Ofenbeck, 2009).

### *Memetic CMA-ES*

Memetic algorithms are algorithms that combine global evolutionary variation operators with local search mechanisms. The term "memetic" is derived from the word *meme*, which has been coined by Richard Dawkins in "The selfish gene" (Dawkins, 1967). A meme is a beneficial cultural "gene" that is not inherited by reproduction, but rather by participation of

the individual in society and culture. In the optimization context, the local search of an individual in the population simulates this process of acquiring beneficial information from its social context (i.e., the local landscape). Memetic algorithms are particularly popular in combinatorial optimization. In the continuous case, memetic algorithms can be used whenever some smoothness about the objective function can be assumed. In principle, local search can be conducted by any suitable local descent method. However, memetic algorithms are often only of practical relevance when analytic gradient information is explicitly available. Efficient first-order optimization methods can then be used for the local search. Such information is often available for model energy landscapes in physics and chemistry. We thus explored the possibility of combining such information into CMA-ES, resulting in a memetic CMA-ES (M-CMA-ES). Due to our focus on black-box methods, we only present the key idea of M-CMA-ES. Details can be found elsewhere (Ofenbeck, 2009; Misteli and Ofenbeck, 2010).

In M-CMA-ES, each candidate solution $\mathbf{x}_k^{(g+1)}$ in the current population is subject to a steepest descent or a Quasi-Newton minimization. The resulting minimizer is denoted $\mathbf{x}_k^{(g+1)\,*}$. This

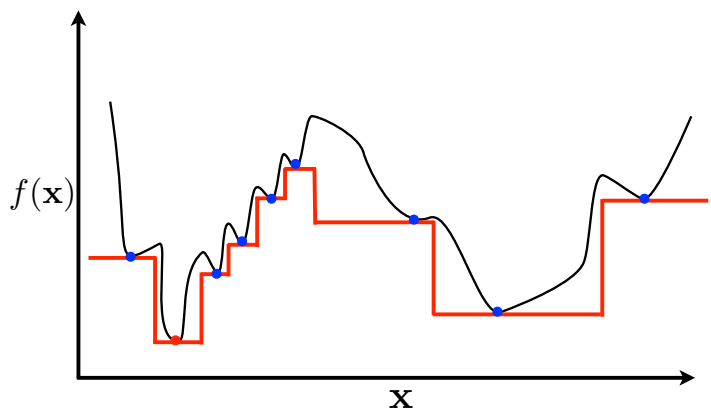



Figure 4.4: Model of a double-funnel landscape with one optimal narrow rugged funnel and a broad sub-optimal funnel. When strict local descent is applied to every search point, the resulting landscape is a double staircase (red) which is much simpler to optimize.

procedure can be understood as a transformation of the fitness landscape (Wales and Doye, 1997) as visualized in Fig. 4.4. Any sequence of neighboring basins with decreasing fitness will be transformed into a staircase, thus removing local multi-modality. Selection is realized by ranking the $\lambda$ sample points in order of ascending fitness of the minimizers $\mathbf{x}_k^{(g+1)\,*}$ rather than the fitness of the initial samples $\mathbf{x}_k^{(g+1)}$. Again, the $\mu$ best individuals are chosen for recombination. Two choices are possible to update the mean of the sampling distribution: In the Baldwinian view of evolution, the information about the locations of the minima are *not* transferred to the next generation, hence Eq. (4.2) is applied. In the Lamarckian view of evolution, the minima locations are inherited by the next generation, leading to weighted

intermediate recombination of the selected minima:

$$\mathbf{m}^{(g+1)} = \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{(g+1)\,*}, \tag{4.13}$$

with standard weights $w_i$. A meaningful hybridization of local descent and CMA-ES can only be achieved if we ensure that the local minimizers are within the likely sampling space of the current multivariate normal distribution. Then, the above averaging makes sense, and CMA-ES can learn the correlation structure of neighboring local minima. For high-dimensional, highly multi-modal landscapes, however, there is no efficient way to achieve this.

The large number of gradient evaluations potentially required by steepest descent methods to arrive at a local minimum can be reduced by Quasi-Newton methods. The large, non-local steps of these methods would, however, invalidate the hybridization with CMA-ES. This can be avoided by restricting Quasi-Newton steps to an ellipsoidal region defined by the current mean and covariance matrix of CMA-ES (Misteli and Ofenbeck, 2010). M-CMA-ES is available in pCMALib using the BFGS module (Nocedal, 1980) as a gradient-based minimizer. M-CMA-ES has been applied to the Lennard-Jones cluster problem (Ofenbeck, 2009).

### 4.2.3 Benchmark results for low-discrepancy CMA-ES

We now quantify the performance of low-discrepancy CMA-ES on the IEEE CEC 2005 benchmark test suite (see Section 2.3.2. The performance of BLR-CMA-ES will be investigated in Chapter 7 in the context of linear chain problems. M-CMA-ES cannot be benchmarked on the present test suite because, for the majority of the problems, analytic gradients are not known or do not exist.

We present three sets of numerical results: (i) the original data for IPOP-CMA-ES produced by Hansen's MATLAB implementation and reported in (Auger and Hansen, 2005b) (Table 4.1), (ii) IPOP-CMA-ES as implemented in pCMALib (Table 4.2), and (iii) low-discrepancy IPOP-CMA-ES as implemented in pCMALIb (Table 4.3). Comparison between the first two tables validates the expected performance of CMA-ES in pCMALib. Note that this comparison serves as a high-level validation of the algorithmic implementation. Further validation scenarios can be found in (Baumgartner, 2008; Müller et al., 2009a). The latter two tables provide a means to analyze the performance of LD sampling.

From the wealth of resulting data we mainly focus on statistics about the number of FES needed to solve a certain problem (see Tables 2.2 and 2.3 for the precise conditions of success). We measure algorithm performance using two orthogonal quantities (Suganthan et al., 2005), the **Success Rate** = **#successful runs**/**#runs** and the **Success Performance** = **mean(FES for successful runs)** $\cdot$ **#runs**/**#successful runs**. The *Success Rate* is an estimator for the probability of success $p_\mathrm{s}$, while the *Success Performance* (Succ. Perf.) is an estimator for the speed of convergence.

**Comparison between MATLAB's and pCMALib's CMA-ES.** For the unimodal functions $f_1$–$f_3$ and the basic multi-modal functions $f_6$ and $f_7$, the performance of the two implementations agrees well across all dimensions. The empirical success rate, the success performance, and as well as min, median, max, and mean FES are reproduced by our implementation up to statistical variations. Larger variations exist for the $f_4$ ($f_2$ with multiplicative noise) in $n = 30$ as well as for $f_5$ in $n = 50$, where our implementation has a non-zero $p_s$. This can be explained by the fact that in the original experiment the starting point for $f_4$ minimizations has been set incorrectly (see erratum at `http://www.lri.fr/~hansen/cec2005ipopcmaes.txt`). In this erratum it is also stated that "On some function the boundary setting was omitted, but this has most probably no impact on the results." Our results on the pair of Rastrigin functions $f_9/f_{10}$ disprove this statement. While the results in Table 4.1 suggest that $f_9/f_{10}$ can be solved even in $n = 50$, we never observed any success with our pCMALib implementation when using the correct boundary settings. This could be reproduced also with Hansen's MATLAB implementation with correct boundary settings, which confirms our observation: IPOP-CMA-ES fails to solve $f_9/f_{10}$ for $n > 15$. Inspection of the best solutions found by IPOP-CMA-ES reveals that in many dimensions the algorithm converges to the optimal position, while it gets stuck at the boundary in the others. The same is true for the performance of IPOP-CMA-ES on $f_{12}$, the highly-multi-modal Schwefel function, in $n = 30$. The success rate of IPOP-CMA-ES also decreases to 0 there when correct boundary settings are used.

Taken together, three main conclusions can be drawn from the numerical data. First, some of the published results of IPOP-CMA-ES are not correct. Second, the current boundary handling mechanism in CMA-ES (Hansen et al., 2009) needs to be revisited in order to efficiently solve constrained Rastrigin-like functions. Third, pCMALib is likely to be a correct implementation of (IPOP-)CMA-ES. The latter conclusion is also supported by the validation scenarios presented in (Baumgartner, 2008).

**Performance of IPOP-CMA-ES with and without LD sequences.** The data in Tables 4.2 and 4.3 provide strong evidence that LD sampling improves the performance of IPOP-CMA-ES. We observe the same or better success rates and performances for all successfully solved functions in all dimensions, except $f_5$ in $n = 30, 50$ dimensions. The minimum of this unimodal function is located at the boundary of the domain, suggesting that for this type of problem, pseudo-random sampling might be better. The set of unimodal functions $f_1$–$f_3$ can be solved faster with LD sampling. The noisy $f_4$ can be solved in all dimensions with $p_s = 1$, while standard IPOP-CMA-ES can only solve it with $p_s = 0.28$ for $n = 30$ and fails in $n = 50$ dimensions. The highly multi-modal Weierstrass function $f_{11}$ can be solved for $n = 10$ with a remarkable rate of $p_s = 0.84$ (compared to $p_s = 0.32$ without LD sampling). LD sampling also enables IPOP-CMA-ES to solve the shifted/rotated Rastrigin $f_{10}$ in $n = 30$, once.
These results confirm our hypothesis that LD sampling leads to faster and more robust performance of CMA-ES. This is also in agreement with Teytaud and Gelly's benchmark results on other test functions using CMA-ES with Halton sequences (Teytaud and Gelly, 2007). The reduced performance on $f_5$, where the global minimum is at the boundary, needs further investigation. Nonetheless, these results lead us to decide that the standard parameter settings in pCMALib use quasi-random sampling with Sobol' sequences and Acklam's inversion rather than PRNG's. We recommend this also for other CMA-ES implementations.

| n=10 | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $\mathbf{p}_s$ | **Succ. Perf.** |
| f1 | 1.44e+03 | 1.63e+03 | 1.71e+03 | 1.61e+03 | 6.14e+01 | 1.00 | 1.61e+03 |
| f2 | 2.20e+03 | 2.35e+03 | 2.60e+03 | 2.38e+03 | 1.06e+02 | 1.00 | 2.38e+03 |
| f3 | 5.84e+03 | 6.51e+03 | 7.20e+03 | 6.50e+03 | 2.92e+02 | 1.00 | 6.50e+03 |
| f4 | 2.52e+03 | 2.88e+03 | 3.22e+03 | 2.90e+03 | 1.68e+02 | 1.00 | 2.90e+03 |
| f5 | 5.36e+03 | 5.83e+03 | 6.72e+03 | 5.85e+03 | 2.89e+02 | 1.00 | 5.85e+03 |
| f6 | 5.67e+03 | 8.55e+03 | 2.26e+04 | 1.08e+04 | 5.00e+03 | 1.00 | 1.08e+04 |
| f7 | 1.49e+03 | 5.83e+03 | 1.33e+04 | 4.67e+03 | 2.83e+03 | 1.00 | 4.67e+03 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 2.33e+04 | 7.85e+04 | - | 5.75e+04 | 2.11e+04 | 0.76 | 7.57e+04 |
| f10 | 2.68e+04 | 5.15e+04 | - | 5.98e+04 | 1.81e+04 | 0.92 | 6.50e+04 |
| f11 | 3.05e+04 | - | - | 6.31e+04 | 2.56e+04 | 0.24 | 2.63e+05 |
| f12 | 2.37e+03 | 3.10e+04 | - | 2.88e+04 | 2.78e+04 | 0.88 | 3.27e+04 |
| n=30 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 4.15e+03 | 4.50e+03 | 4.72e+03 | 4.50e+03 | 1.33e+02 | 1.00 | 4.50e+03 |
| f2 | 1.20e+04 | 1.31e+04 | 1.36e+04 | 1.30e+04 | 3.52e+02 | 1.00 | 1.30e+04 |
| f3 | 4.15e+04 | 4.27e+04 | 4.42e+04 | 4.27e+04 | 6.06e+02 | 1.00 | 4.27e+04 |
| f4 | 1.94e+04 | - | - | 2.36e+04 | 4.79e+03 | 0.40 | 5.90e+04 |
| f5 | 1.91e+04 | 6.83e+04 | 1.03e+05 | 6.59e+04 | 1.85e+04 | 1.00 | 6.59e+04 |
| f6 | 3.76e+04 | 4.83e+04 | 1.55e+05 | 6.00e+04 | 2.81e+04 | 1.00 | 6.00e+04 |
| f7 | 4.12e+03 | 4.97e+03 | 1.99e+04 | 6.11e+03 | 4.02e+03 | 1.00 | 6.11e+03 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 2.75e+05 | - | - | 2.85e+05 | 6.87e+03 | 0.36 | 7.90e+05 |
| f10 | 2.87e+05 | - | - | 2.90e+05 | 2.44e+03 | 0.12 | 2.42e+06 |
| f11 | 1.99e+05 | - | - | 1.99e+05 | 0.00e+00 | 0.04 | 4.98e+06 |
| f12 | 1.67e+04 | - | - | 7.19e+04 | 7.54e+04 | 0.32 | 2.25e+05 |
| n=50 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 6.54e+03 | 6.89e+03 | 7.13e+03 | 6.88e+03 | 1.42e+02 | 1.00 | 6.88e+03 |
| f2 | 3.00e+04 | 3.11e+04 | 3.29e+04 | 3.13e+04 | 6.55e+02 | 1.00 | 3.13e+04 |
| f3 | 1.15e+05 | 1.17e+05 | 1.18e+05 | 1.17e+05 | 6.77e+02 | 1.00 | 1.17e+05 |
| f4 | - | - | - | - | - | 0.00 | - |
| f5 | - | - | - | - | - | 0.00 | - |
| f6 | 1.15e+05 | 1.36e+05 | 3.59e+05 | 1.58e+05 | 6.68e+04 | 1.00 | 1.58e+05 |
| f7 | 7.32e+03 | 8.00e+03 | 1.01e+04 | 8.03e+03 | 5.56e+02 | 1.00 | 8.03e+03 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 4.06e+05 | - | - | 4.35e+05 | 2.22e+04 | 0.28 | 1.55e+06 |
| f10 | 4.32e+05 | - | - | 4.52e+05 | 2.00e+04 | 0.12 | 3.76e+06 |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | - | - | - | - | - | 0.00 | - |

Table 4.1: Number of FES (min, median, maximum, mean, and standard deviation) for IPOP-CMA-ES to reach $f(\mathbf{x}_{\min}) + \epsilon$ as measured by (Auger and Hansen, 2005b). We show the results for $f_1$–$f_{12}$ in $n = 10, 30, 50$ within MAX_FES $= 10^4 n$. The results for $f_{13}$–$f_{25}$ are not shown since none of these functions could ever be solved by IPOP-CMA-ES. The second-to-last column shows the empirical success rates $p_s$, the last column the success performance.

| n=10 | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $\mathbf{p}_s$ | **Succ. Perf.** |
| f1 | 1.55e+03 | 1.69e+03 | 1.85e+03 | 1.68e+03 | 6.72e+01 | 1.00 | 1.68e+03 |
| f2 | 2.19e+03 | 2.38e+03 | 2.53e+03 | 2.38e+03 | 1.08e+02 | 1.00 | 2.38e+03 |
| f3 | 6.23e+03 | 6.57e+03 | 7.36e+03 | 6.63e+03 | 3.11e+02 | 1.00 | 6.63e+03 |
| f4 | 2.24e+03 | 2.54e+03 | 2.85e+03 | 2.56e+03 | 1.56e+02 | 1.00 | 2.56e+03 |
| f5 | 5.21e+03 | 5.69e+03 | 6.46e+03 | 5.76e+03 | 3.20e+02 | 1.00 | 5.76e+03 |
| f6 | 5.36e+03 | 7.54e+03 | 3.54e+04 | 1.05e+04 | 6.87e+03 | 1.00 | 1.05e+04 |
| f7 | 1.38e+03 | 2.05e+03 | 1.26e+04 | 4.11e+03 | 3.20e+03 | 1.00 | 4.11e+03 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 4.64E+04 | 8.27E+04 | - | 7.09E+04 | 1.71E+04 | 0.80 | 8.87E+04 |
| f10 | 2.47E+04 | 8.16E+04 | - | 6.82E+04 | 2.26E+04 | 0.80 | 8.52E+04 |
| f11 | 2.63E+04 | - | - | 5.86E+04 | 2.39E+04 | 0.32 | 1.83E+05 |
| f12 | 2.52E+03 | 5.37E+04 | - | 3.23E+04 | 3.34E+04 | 0.72 | 4.48E+04 |
| n=30 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 4.41e+03 | 4.68e+03 | 5.14e+03 | 4.70e+03 | 2.10e+02 | 1.00 | 4.70e+03 |
| f2 | 1.23e+04 | 1.28e+04 | 1.33e+04 | 1.27e+04 | 2.81e+02 | 1.00 | 1.27e+04 |
| f3 | 4.19e+04 | 4.37e+04 | 4.46e+04 | 4.36e+04 | 6.82e+02 | 1.00 | 4.36e+04 |
| f4 | 2.02e+04 | - | - | 6.20e+04 | 9.21e+04 | 0.28 | 2.21e+05 |
| f5 | 1.94e+04 | 6.51e+04 | 8.99e+04 | 6.57e+04 | 1.66e+04 | 1.00 | 6.57e+04 |
| f6 | 3.31e+04 | 4.99e+04 | 1.40e+05 | 6.10e+04 | 2.92e+04 | 1.00 | 6.10e+04 |
| f7 | 4.53e+03 | 5.03e+03 | 1.86e+04 | 6.64e+03 | 4.39e+03 | 1.00 | 6.64e+03 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | - | - | - | - | - | 0.00 | - |
| f10 | - | - | - | - | - | 0.00 | - |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | - | - | - | - | - | 0.00 | - |
| n=50 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 6.86e+03 | 7.35e+03 | 8.72e+03 | 7.50e+03 | 4.25e+02 | 1.00 | 7.50e+03 |
| f2 | 2.94e+04 | 3.09e+04 | 3.16e+04 | 3.09e+04 | 5.28e+02 | 1.00 | 3.09e+04 |
| f3 | 1.15e+05 | 1.17e+05 | 1.20e+05 | 1.18e+05 | 1.21e+03 | 1.00 | 1.18e+05 |
| f4 | - | - | - | - | - | 0.00 | - |
| f5 | 4.99e+05 | - | - | 5.00e+05 | 6.71e+00 | 0.20 | 2.50e+06 |
| f6 | 8.88e+04 | 1.30e+05 | 3.14e+05 | 1.49e+05 | 6.01e+04 | 1.00 | 1.49e+05 |
| f7 | 7.26e+03 | 8.01e+03 | 3.08e+04 | 1.08e+04 | 7.35e+03 | 1.00 | 1.08e+04 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | - | - | - | - | - | 0.00 | - |
| f10 | - | - | - | - | - | 0.00 | - |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | - | - | - | - | - | 0.00 | - |

Table 4.2: Number of FES (min, median, maximum, mean, and standard deviation) for IPOP-CMA-ES of **pCMALib** to reach $f(\mathbf{x}_{\min}) + \epsilon$. We show the results for $f_1$–$f_{12}$ in $n = 10, 30, 50$ within MAX_FES $= 10^4 n$. The results for $f_{13}$–$f_{25}$ are not shown since none of these functions could ever be solved by IPOP-CMA-ES. The second-to-last column shows the empirical success rates $p_s$, the last column the success performance.

| n=10 | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $\mathbf{p}_s$ | **Succ. Perf.** |
| f1 | 1.25e+03 | 1.38e+03 | 1.53e+03 | 1.37e+03 | 6.08e+01 | 1.00 | 1.37e+03 |
| f2 | 1.80e+03 | 1.97e+03 | 2.06e+03 | 1.96e+03 | 7.70e+01 | 1.00 | 1.96e+03 |
| f3 | 5.23e+03 | 5.46e+03 | 6.34e+03 | 5.52e+03 | 2.29e+02 | 1.00 | 5.52e+03 |
| f4 | 1.84e+03 | 2.08e+03 | 2.37e+03 | 2.08e+03 | 1.29e+02 | 1.00 | 2.08e+03 |
| f5 | 4.40e+03 | 4.78e+03 | 5.21e+03 | 4.79e+03 | 1.77e+02 | 1.00 | 4.79e+03 |
| f6 | 5.01e+03 | 9.02e+03 | 1.79e+04 | 1.00e+04 | 4.55e+03 | 1.00 | 1.00e+04 |
| f7 | 1.17e+03 | 1.44e+03 | 4.91e+03 | 2.94e+03 | 1.75e+03 | 1.00 | 2.94e+03 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 1.85e+04 | 3.69e+04 | - | 4.21e+04 | 1.75e+04 | 0.88 | 4.78e+04 |
| f10 | 9.99e+03 | 3.74e+04 | - | 4.49e+04 | 1.72e+04 | 0.96 | 4.68e+04 |
| f11 | 9.83e+03 | 4.75e+04 | - | 4.45e+04 | 2.69e+04 | 0.84 | 5.29e+04 |
| f12 | 2.17e+03 | 1.31e+04 | - | 1.76e+04 | 1.79e+04 | 0.88 | 2.00e+04 |
| n=30 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 3.55e+03 | 3.67e+03 | 3.80e+03 | 3.67e+03 | 7.51e+01 | 1.00 | 3.67e+03 |
| f2 | 1.06e+04 | 1.10e+04 | 1.17e+04 | 1.11e+04 | 2.85e+02 | 1.00 | 1.11e+04 |
| f3 | 3.73e+04 | 3.94e+04 | 4.07e+04 | 3.93e+04 | 6.68e+02 | 1.00 | 3.93e+04 |
| f4 | 3.55e+04 | 3.92e+04 | 7.46e+04 | 5.30e+04 | 1.69e+04 | 1.00 | 5.30e+04 |
| f5 | 1.57e+04 | 2.43e+05 | - | 2.34e+05 | 5.72e+04 | 0.92 | 2.55e+05 |
| f6 | 3.77e+04 | 4.91e+04 | 1.27e+05 | 5.85e+04 | 2.61e+04 | 1.00 | 5.85e+04 |
| f7 | 3.62e+03 | 4.04e+03 | 3.02e+04 | 5.56e+03 | 5.59e+03 | 1.00 | 5.56e+03 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | - | - | - | - | - | 0.00 | - |
| f10 | 2.12e+05 | - | - | 2.12e+05 | 0.00e+00 | 0.04 | 5.30e+06 |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | - | - | - | - | - | 0.00 | - |
| n=50 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 5.57e+03 | 5.85e+03 | 6.75e+03 | 5.95e+03 | 2.85e+02 | 1.00 | 5.95e+03 |
| f2 | 2.63e+04 | 2.72e+04 | 2.80e+04 | 2.73e+04 | 4.83e+02 | 1.00 | 2.73e+04 |
| f3 | 1.07e+05 | 1.09e+05 | 1.12e+05 | 1.09e+05 | 1.11e+03 | 1.00 | 1.09e+05 |
| f4 | 1.84e+05 | 1.94e+055 | 2.09e+05 | 1.94e+05 | 5.34e+03 | 1.00 | 1.94e+05 |
| f5 | - | - | - | - | - | 0.00 | - |
| f6 | 1.11e+05 | 1.19e+05 | 2.78e+05 | 1.32e+05 | 4.34e+04 | 1.00 | 1.32e+05 |
| f7 | 5.64e+03 | 6.18e+03 | 8.01e+03 | 6.42e+03 | 5.96e+02 | 1.00 | 6.42e+03 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | - | - | - | - | - | 0.00 | - |
| f10 | - | - | - | - | - | 0.00 | - |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | - | - | - | - | - | 0.00 | - |

Table 4.3: Number of FES (min, median, maximum, mean, and standard deviation) for **low-discrepancy** IPOP-CMA-ES of **pCMALib** to reach $f(\mathbf{x}_{\min}) + \epsilon$. We show the results for $f_1$–$f_{12}$ in $n = 10, 30, 50$ within MAX_FES $= 10^4 n$. The results for $f_{13}$–$f_{25}$ are not shown since none of these functions could ever be solved by IPOP-CMA-ES. The second-to-last column shows the empirical success rates $p_s$, the last column the success performance.

## 4.3 Parallel CMA-ES

CMA-ES has been designed for efficient search on landscapes with a single-funnel topology. On multi-funnel landscapes, the only mechanism to escape from sub-optimal funnels is to restart at a random position in the landscape. This iterated local search idea is well known in the field of heuristic search. Both for combinatorial and continuous problems, it is often considered the only practical solution. For black-box optimization, we offer an alternative approach here: *Parallel CMA-ES*. The concept of parallel CMA-ES is to simultaneously explore the landscape with multiple CMA-ES instances. Figure 4.5 sketches this general idea. In the evolutionary

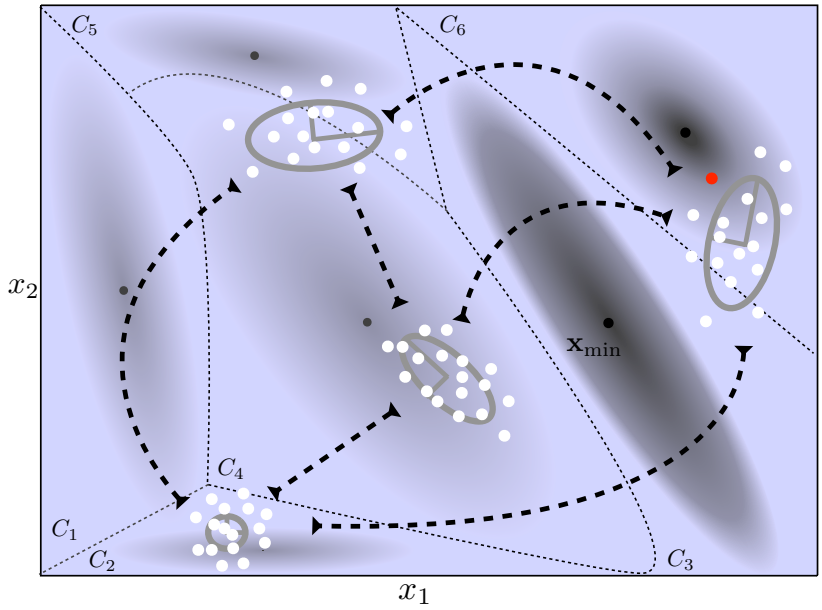



Figure 4.5: Sketch of a parallel CMA-ES scheme. Multiple CMA-ES runs explore a model landscape with 6 basins $C_i$, $i = 1, \ldots, 6$. $C_3$ contains the global minimum. The red individual of the CMA-ES instance in $C_6$ denotes the currently best solution. The black dashed lines indicate that all CMA-ES instances can exchange information about their current states.

computation community, this concept of parallel evolution of several populations is referred to as "coarsely grained parallel" or parallel island model (see (Alba, 2005) for a summary on parallel search heuristics). In parallel CMA-ES, little might be gained by *independent* parallel landscape exploration of $S$ CMA-ES instances. When an unlimited budget of FES is available, such a scheme is equivalent to $S$ sequential CMA-ES runs with identical starting points and the same sequence of pseudo-/quasi-random numbers. An essential ingredient in parallel CMA-ES is the exchange of information between the different populations. Each CMA-ES instance explores a different part of the landscape and communicates certain landscape characteristics and/or internal state variables to other CMA-ES instances. This global information can be used by each instance for improving its individual search performance. Different paradigms are conceivable. For instance, if a large coverage of the landscape is desirable, the different

instances can be dynamically forced to explore disjoints parts of the search space, reminiscent to the classical tabu search paradigm (Glover, 1989). We can also adopt an approximate branch-and-bound scheme where different parts of the landscape are explored in parallel and their current fitness is globally communicated. Landscape regions that yield worse solutions than other, already explored, parts are discarded, and search proceeds in other directions. Such a scheme can, however, only be approximate because the usual complete enumeration of the search space is not feasible in a black-box setting. We concretize this idea in *Particle Swarm CMA-ES*.

### 4.3.1 Particle Swarm CMA-ES

Particle Swarm CMA-ES (PS-CMA-ES) extends the canonical CMA-ES by collaborative concepts from Particle Swarm Optimization (PSO) (Kennedy and Eberhart, 1995). PSO is a population-based optimization method in which particles "fly" over the search landscape and employ a *cooperative* strategy to move toward one another based on knowledge of local *and* global best particle positions. PS-CMA-ES combines the robust *local* search performance of CMA-ES with the *global* exploration power of PSO, using multiple ($S$) CMA-ES instances to explore different parts of the landscape in parallel. Swarm intelligence is introduced by considering individual CMA-ES instances as lumped particles that communicate with each other. This includes non-local information in individual CMA-ES instances, which influences their search directions and positions. We draw some inspiration from PSO, which will be introduced first. We then describe the algorithm, provide numerical performance measurements, and present recent developments and future research opportunities.

#### *Particle Swarm Optimization*

Each particle in standard PSO is described by its position $\mathbf{p} \in \mathbb{R}^n$ and its velocity $\mathbf{v} \in \mathbb{R}^n$ (Kennedy and Eberhart, 1995). While moving through search space, a swarm of particles evaluates the fitness function and updates its velocity according to an update rule that incorporates both local and global information. The local information of each particle is given by the best solution this particle has found so far ($\mathbf{p}_{\mathrm{l,best}}$). The global information corresponds to the best solution any member of the swarm has found so far ($\mathbf{p}_{\mathrm{g,best}}$). The velocities of all particles at time $t+1$ are updated from the old velocities and the positions at time $t$ as:

$$\mathbf{v}^{(t+1)} = \mathbf{v}^{(t)} + c_1 r_1 \left( \mathbf{p}^{(t)}_{\mathrm{l,best}} - \mathbf{p}^{(t)} \right) + c_2 r_2 \left( \mathbf{p}^{(t)}_{\mathrm{g,best}} - \mathbf{p}^{(t)} \right) . \tag{4.14}$$

The new positions are computed using an explicit Euler integrator with a time step of 1, thus:

$$\mathbf{p}^{(t+1)} = \mathbf{p}^{(t)} + \mathbf{v}^{(t+1)} . \tag{4.15}$$

The uniform random numbers $r_1$, $r_2 \in [0, 1]$ introduce a stochastic element to the algorithm, and $c_1$ and $c_2$ weight the influence of local vs. global information (Kennedy and Eberhart, 1995). Since its inception in 1995, thousands of articles on PSO have been published. We refer to (Banks et al., 2007) for a recent review. A physical view on PSO, linking the algorithm to Newtonian mechanics, is provided in (Mikki and Kishk, 2007).

### The PS-CMA-ES algorithm

We hypothesize that exchange of information between parallel CMA-ES runs enhances the performance of CMA-ES on multi-modal functions, in particular functions with multiple funnels. Global swarm information is included both in the covariance adaptation mechanism of CMA-ES and in the placement of the population mean. Each CMA-ES instance already exploits local information. We set $c_1 = 0$ in PS-CMA-ES. The parameters $c_2$ and $r_2$ are replaced by a more elaborate weighting rule.

**Adapting the covariance matrix.** We adjust the covariance matrix of CMA-ES such that it is more likely to sample good candidates in direction of the current global best position $\mathbf{p}_{\mathrm{g,best}} \in \mathbb{R}^n$ in the swarm. This is achieved by mixing the CMA covariance matrix from Eq. (4.9) with a PSO covariance matrix that is influenced by global information:

$$\mathbf{C}^{(g+1)} = c_{\mathrm{p}}\, \mathbf{C}_{\mathrm{CMA}}^{(g+1)} + (1 - c_{\mathrm{p}})\, \mathbf{C}_{\mathrm{PSO}}^{(g+1)}\,, \tag{4.16}$$

where the mixing weight $c_{\mathrm{p}} \in [0, 1]$ is a new strategy parameter. $\mathbf{C}_{\mathrm{CMA}}^{(g+1)}$ follows the original adaptation rule as given in Eq. (4.9). $\mathbf{C}_{\mathrm{PSO}}^{(g+1)}$ is a rotated version of $\mathbf{C}_{\mathrm{CMA}}^{(g)}$ such that the largest eigenvector $\mathbf{b}_{\mathrm{main}}$ of $\mathbf{C}_{\mathrm{CMA}}^{(g)}$ is aligned with the vector $\mathbf{p}_{\mathrm{g}} = \mathbf{p}_{\mathrm{g,best}} - \mathbf{m}^{(g)}$ that points from the current mean $\mathbf{m}^{(g)}$ toward the global best position $\mathbf{p}_{\mathrm{g,best}}$. The vector $\mathbf{b}_{\mathrm{main}}$ hence is the analog of the velocity direction vector in PSO. $\mathbf{C}_{\mathrm{CMA}}^{(g)}$ can be decomposed as $\mathbf{C}_{\mathrm{CMA}}^{(g)} = \mathbf{B}\,\mathbf{D}^2\,\mathbf{B}^T$, such that the rotated covariance matrix can be constructed by rotating its eigenvectors (columns of $\mathbf{B}$). This yields the orthogonal matrix $\mathbf{B}_{\mathrm{rot}}^{(g)} = \mathbf{R}\,\mathbf{B} \in \mathbb{R}^{n\times n}$ of the rotated eigenvectors. $\mathbf{C}_{\mathrm{PSO}}^{(g+1)}$ is then given by:

$$\mathbf{C}_{\mathrm{PSO}}^{(g+1)} = \mathbf{B}_{\mathrm{rot}}^{(g)} \left(\mathbf{D}^{(g)}\right)^2 \left(\mathbf{B}_{\mathrm{rot}}^{(g)}\right)^T. \tag{4.17}$$

The rotation matrix $\mathbf{R} \in \mathbb{R}^{n\times n}$ is uniquely and efficiently computed using *Givens Rotations* (Golub and Van Loan, 1996). The Givens rotation matrix $\mathbf{G}$ describes a unique rotation of a vector onto one axis. An $n$-dimensional rotation is performed as a sequence of two-dimensional rotations for all possible pairs $(i, j)$ of axes (Rudolph, 1992):

$$\mathbf{R}^{(n\times n)} = \prod_{i=1}^{n-1} \prod_{j=i+1}^{n} \mathbf{R}_{ij}\,. \tag{4.18}$$

$\mathbf{R}_{ij}$ is an $n \times n$ matrix describes the unique rotation in the plane spanned by the axes $(i, j)$. It can be considered as a rank-two correction to the identity:

$$\mathbf{R}_{ij,\mathbf{R}_{\mathrm{plane}}} = \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & \dots & 0 & \dots & 0 & \dots & 0 \end{array} \\ \begin{array}{ccccccc} \vdots & \ddots & \vdots & & \vdots & & \vdots \end{array} \\ \begin{array}{ccccccc} 0 & \dots & \mathbf{R}_{\mathrm{plane}}(1,1) & \dots & \mathbf{R}_{\mathrm{plane}}(1,2) & \dots & 0 \end{array} \\ \begin{array}{ccccccc} \vdots & & \vdots & \ddots & \vdots & & \vdots \end{array} \\ \begin{array}{ccccccc} 0 & \dots & \mathbf{R}_{\mathrm{plane}}(2,1) & \dots & \mathbf{R}_{\mathrm{plane}}(2,2) & \dots & 0 \end{array} \\ \begin{array}{ccccccc} \vdots & & \vdots & & \vdots & \ddots & \vdots \end{array} \\ \begin{array}{ccccccc} 0 & \dots & 0 & \dots & 0 & \dots & 1 \end{array} \\ \begin{array}{ccccccc} 1 & & i & & j & & n \end{array} \end{array} \begin{array}{l} 1 \\ \\ i \\ \\ j \\ \\ n \end{array}, \tag{4.19}$$

where the $2 \times 2$ matrix $\mathbf{R}_{\text{plane}} = \mathbf{G}_{\text{p}}^T \, \mathbf{G}_{\text{b}}$. $\mathbf{G}_{\text{p}}$ is the Givens rotation of the elements $i$ and $j$ of $\mathbf{p}_{\text{g}}$, and $\mathbf{G}_{\text{b}}$ is the Givens rotation of the elements $i$ and $j$ of $\mathbf{b}_{\text{main}}$. The complete procedure to compute the rotation matrix $\mathbf{R} \in \mathbb{R}^{n \times n}$ is summarized in Algorithm 1.

**Algorithm 1: Efficient computation of the $n$-dimensional rotation matrix using Givens rotations**

**Input**: Two $n$-dimensional vectors $\mathbf{b}_{\text{main}}$ and $\mathbf{p}_{\text{g}}$
**Result**: Rotation matrix $\mathbf{R}$, such that $\mathbf{R}\,\mathbf{b}_{\text{main}} = a\,\mathbf{p}_{\text{g}}$
**Initialization:** $\mathbf{p} = \mathbf{p_g}$ , $\mathbf{b} = \mathbf{b_{main}}$
**for** $i = (n-1), -1, 1$ **do**
  **for** $j = n, -1, (i+1)$ **do**

$$\mathbf{p_{plane}} = \begin{pmatrix} p(i) \\ p(j) \end{pmatrix} \qquad \mathbf{b_{plane}} = \begin{pmatrix} b(i) \\ b(j) \end{pmatrix}$$

$$\mathbf{G_p} = \mathbf{Givens}(\mathbf{p_{plane}}) \qquad \mathbf{G_b} = \mathbf{Givens}(\mathbf{b_{plane}})$$

$$\mathbf{p} \leftarrow \mathbf{R}_{ij,\mathbf{G_p}}\,\mathbf{p} \qquad \mathbf{b} \leftarrow \mathbf{R}_{ij,\mathbf{G_b}}\,\mathbf{b}$$

$$\mathbf{R_p} \leftarrow \mathbf{R}_{ij,\mathbf{G_p}}\,\mathbf{R_p} \qquad \mathbf{R_b} \leftarrow \mathbf{R}_{ij,\mathbf{G_b}}\,\mathbf{R_b}$$

  **end**
**end**
$\mathbf{R} = \mathbf{R_p}^T \mathbf{R_b}$

**Biasing the mean value.** In order to enable individual swarm particles (i.e., CMA-ES instances) in PS-CMA-ES to escape from local minima, we bias the mean value in addition to rotating the covariance matrix in direction of the global best solution. After the recombination step of each CMA-ES generation, the updated mean value for the next generation $g+1$ is biased as:

$$\mathbf{m}^{(g+1)} \leftarrow \mathbf{m}^{(g+1)} + \mathbf{bias}\,. \tag{4.20}$$

The bias changes the evolution path $\mathbf{p}_c^{(g+1)}$ update for future generations, since the path will be computed with respect to the biased mean value. The biasing rules can be used to include prior knowledge about the structure of the problem, e.g., the correlation length of the fitness landscape. In our benchmark tests, we discriminate the following 3 exploration scenarios that are coupled to the step size $\sigma$ of each individual CMA-ES instance, a natural measure for its mode of exploration:

1. The CMA-ES instance that produced $\mathbf{p}_{\text{g,best}}$ and all CMA-ES instances with step sizes $\sigma > ||\mathbf{p}_{\text{g}}||$ are not biased at all, thus: $\mathbf{bias} = 0$. Using no bias in these cases avoids catapulting the global best member out of the funnel containing the potential global optimum and prevents very explorative runs from overshooting the target.

2. CMA-ES instances that have converged to local minima far from $\mathbf{p}_{\text{g,best}}$ are characterized by a ratio $\sigma / ||\mathbf{p}_{\text{g}}||$ that is smaller than $t_{\text{c}} ||\mathbf{p}_{\text{g}}||$, $t_{\text{c}} < 1$. These instances are strongly biased

in order to allow them to escape from the current funnel, thus: $\mathbf{bias} = b\,\mathbf{p}_\mathrm{g}$. Using a large bias prevents these instances from converging back into the same local minimum again.

3. CMA-ES instances that are not converged, and thus still exploring the space, are given a bias equal to the step-size- to-distance ratio: $\mathbf{bias} = \sigma/||\mathbf{p}_\mathrm{g}||\,\mathbf{p}_\mathrm{g}$. Using such a small bias prevents clustering of swarm members and preserves the explorative power of the method.

This set of biasing rules, summarized in Algorithm 2, introduces two new strategy parameters: the convergence threshold $t_\mathrm{c}$ and the biasing factor $b$.

**Algorithm 2**: **Standard rules for biasing the mean value**

**Input**: Mean value $\mathbf{m}^{(g+1)}$ and global best direction $\mathbf{p}_\mathrm{g}$
**Result**: Biased mean value $\mathbf{m}^{(g+1)}$
**if** $\sigma < ||\mathbf{p}_g||$ *and instance has not produced* $\mathbf{p}_g$ **then**
  **if** $\frac{\sigma}{||\mathbf{p}_g||} \leq t_c||\mathbf{p}_g||$ **then**
    $\mathbf{bias} = b\,\mathbf{p}_\mathrm{g}$
  **else**
    $\mathbf{bias} = \frac{\sigma}{||\mathbf{p}_\mathrm{g}||}\,\mathbf{p}_\mathrm{g}$
  **end**
**else**
  $\mathbf{bias} = 0$
**end**
$\mathbf{m}^{(g+1)} = \mathbf{m}^{(g+1)} + \mathbf{bias}$

**Strategy parameters.** In PS-CMA-ES, all swarm members communicate with each other. A swarm of size $S$ thus requires $S(S-1)$ communications. The PSO update (broadcasting the global best solution, rotating the covariance matrices, and biasing the mean values of all CMA instances) must, however, not be performed at each iteration. Too frequent updates would prevent the CMA-ES instances from learning the local covariance matrix. Too infrequent updates would lead to premature convergence, with several CMA-ES instances stopping in local minima before the first swarm information is exchanged. Clearly, a problem-specific tradeoff has to be found for the communication interval $I_\mathrm{c}$ between PSO updates, constituting the main strategy parameter of the PS-CMA-ES. In the limit $I_\mathrm{c} \to \infty$, PS-CMA-ES is equivalent to $S$ parallel standard CMA-ES runs.

Other strategy parameters are the swarm size $S$, the biasing parameters $t_\mathrm{c}$ and $b$, and the mixing weight $c_\mathrm{p}$ in Eq. (4.16). Both $t_\mathrm{c}$ and $b$ have been considered random variables at first, but the setting $t_\mathrm{c} = 0.1$ and $b = 0.5$ was found a more robust choice on most test functions. The weight $c_\mathrm{p}$ can, e.g., be randomized, self-adapted, or determined by a preliminary grid search. The swarm size could be chosen to reflect the dimensionality of the problem or also

using a grid search. It determines the overall population size and the computational overhead of the algorithm. Therefore, $S$ should be chosen as low as possible, but as high as necessary to significantly increase exploration power. For $S = 1$, PS-CMA-ES is equivalent to standard CMA-ES.

**Restart strategies.** It is possible that some CMA-ES instances converge to local minima before reaching the next time point of global communication. There are several possibilities how to handle these instances. A generic idea is to restart these CMA-ES instances at random locations in the landscape. In most cases, however, we are faced with a limited budget of FES. This implies that the restarted CMA-ES instances consume FES that are then not available any more to better CMA-ES instances. The standard setting in PS-CMA-ES thus is to idle all converged CMA-ES runs until *every* swarm member has converged. If the FES budget is not exhausted at this point, all swarm members are restarted. This strategy introduces a second level of competition: Runs that evolve faster toward a low-lying landscape region are allowed to consume more FES.

### 4.3.2 Numerical results and comparison to related algorithms

**Determination of standard strategy parameters** We use grid search to determine good values for the strategy parameters introduced in Section 4.3.1. This determination of strategy parameters can be considered as an unbiased *training* of the algorithm. The strategy parameters of individual CMA-ES instances are set according to (Hansen, 2008). Pseudo-random numbers are used for sampling. The initial step-size $\sigma$ is varied between 20% and 50% of the constrained region of the minimization problem. Table 4.4 summarizes all strategy parameters tested in the grid search. All 144 combinations are tested for functions $f_1$–$f_{25}$ in $n = 10$ dimensions. The benchmark requires 25 repetitions per problem (Suganthan et al., 2005), leading to a total of $25 \cdot 25 \cdot 144 = 90\,000$ PS-CMA-ES runs in total.

| Parameter | Tested values |
|---|---|
| step size $\sigma$ | 0.2, 0.3, 0.4, 0.5 |
| swarm size $S$ | 6, 10, 15 |
| mixing weight $c_\mathrm{p}$ | 0.3, 0.5, 0.7 |
| communication interval $I_\mathrm{c}$ | 150, 200, 250, $\infty$ |

Table 4.4: Values of the PS-CMA-ES strategy parameters tested in the grid search.

The grid search revealed an initial step size of $\sigma = 0.2$ to be the most beneficial one. A swarm size of at least $S = 10$, a mixing weight of $c_\mathrm{p} = 0.5$ or 0.7, and swarm updates every 200 to 250 generations performed well on the test functions. Although there might be better configurations for individual test cases, we consider $\sigma = 0.2$, $S = 16$, $c_\mathrm{p} = 0.7$, and $I_\mathrm{c} = 200$ the standard setting of PS-CMA-ES. In order to test the *generality* of this parameter set, we use these values also for tests in 30 and 50 dimensions.

**Performance of PS-CMA-ES.** As in the previous section, we present the success tables of PS-CMA-ES with and without LD sampling. Because the majority of functions that are highly multi-modal/multi-funnel cannot be solved within the given FES budget we use the mean (over all 25 repetitions of each problem) function value error after MAX_FES=$10^4 n$ as a measure of algorithmic performance. Using this measure, we compare the performance of PS-CMA-ES to that of LR-CMA-ES and IPOP-CMA-ES, as well as to the performance of the Particle Swarm Guided ES (PSGES) (Hsieh et al., 2007), a method that shares similar design principles with PS-CMA-ES. We take the performance results of the reference algorithms from the corresponding original publications (Auger and Hansen, 2005a,b; Hsieh et al., 2007). PSGES data are only available for $n = 10$.

We first analyze Table 4.5 for PS-CMA-ES without LD sampling. For $n = 10$, PS-CMA-ES successfully solves 11 test functions. Among all algorithms ever tested on this benchmark, only IPOP-CMA-ES is able to solve the same number of functions. PS-CMA-ES fails to solve $f_3$, a shifted ellipsoid with high condition number. It also needs one order of magnitude more function evaluations than IPOP-CMA-ES to solve $f_1$, $f_2$, and $f_4$–$f_7$. These results are expected since for smooth unimodal landscape topologies, parallel CMA-ES is not needed. However, PS-CMA-ES outperforms IPOP-CMA-ES on the Rastrigin pair $f_9/f_{10}$ in terms of both success rate and success performance. This strong result implies that even on highly multi-modal landscapes with single-funnel topology, collaborative CMA-ES instances are meaningful. The performance on $f_{11}/f_{12}$ is comparable to that of IPOP-CMA-ES. PS-CMA-ES, however, is the only CMA-ES variant that solves the multi-funnel, hybrid composition function $f_{15}$ (Fig. 4.6). To date, there are only three other algorithms that can also solve this problem (see (Hansen, 2006) for summary statistics of all tested algorithms). It is also noteworthy that alternative parameter settings, tested during the grid search, enabled PS-CMA-ES to solve 12 test problems in $n = 10$, including the hybrid composition function $f_{16}$, the rotated version of $f_{15}$. To our knowledge, there has been no other algorithm so far that solved any of the functions $f_{16}$–$f_{25}$. For $n = 30$, PS-CMA-ES can solve $f_1/f_2$, and $f_7$ with $p_s = 1$, and $f_7$, $f_9/f_{10}$ with $p_s = 0.04$. These results are outperformed by IPOP-CMA-ES, except for the pair $f_9/f_{10}$ that cannot be solved by IPOP-CMA-ES when boundaries are set correctly. In $n = 50$ dimensions, PS-CMA-ES only solves $f_1$ and $f_7$.

The observed benefits of LD sampling in IPOP-CMA-ES can be transferred to PS-CMA-ES. Table 4.6 summarizes the results for PS-CMA-ES with identical settings and LD sampling. In $n = 10$ dimensions, this PS-CMA-ES can solve 12 test functions, including the high-conditional ellipsoid $f_3$ ($p_s = 0.32$). All other functions are solved faster and with equal or higher success rate than when using pseudo-random numbers. These results constitute the best performance among all algorithms ever tested on the CEC 2005 benchmark (Hansen, 2006). In $n = 30$ dimensions, LD sampling enables PS-CMA-ES to solve $f_1/f_2$ and$f_7$with $p_s = 1.0$. Function $f_5$ cannot be solved with this scheme, in contrast to PS-CMA-ES with pseudo-random numbers. However, $f_9/f_{10}$ can be solved with higher success rate, as well as $f_{12}$. Only two other algorithms could solve $f_{12}$ in this dimension (Hansen, 2006); IPOP-CMA-ES with correct boundary settings fails. For $n = 50$, LD sampling allows solving the pair $f_9/f_{10}$ at least once, in addition to $f_1$, $f_2$, and $f_7$.

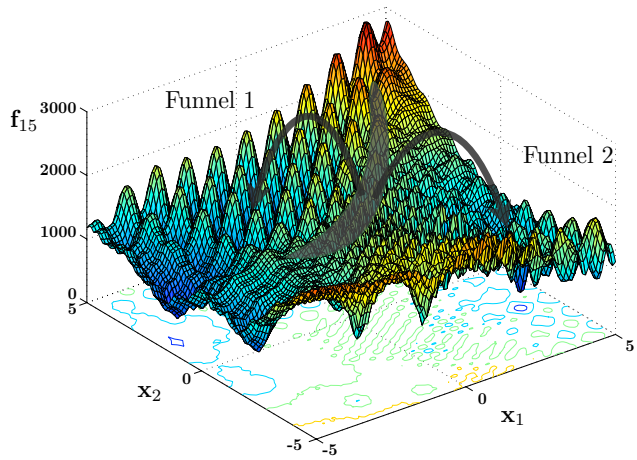



Figure 4.6: Two-dimensional version of the function $f_{16}$ from the CEC benchmark test suite (Suganthan et al., 2005). The global topology is a double funnel separated by the central ridge region (in gray). The global and several local minima are contained in funnel 1, several deep local minima in funnel 2. This topology is hard since a search heuristic can be trapped in the broad funnel 2.

The CEC 2005 benchmark results also confirm that PS-CMA-ES inherits several invariance properties from CMA-ES. Although $f_3$ suggests that collaborative CMA-ES instances are less favorable for highly stretched unimodal topologies, PS-CMA-ES shows no decrease in performance on the pair $f_9/f_{10}$, where $f_{10}$ is a rotated non-separable version of $f_9$. In fact, PS-CMA-ES performs even better on $f_{10}$ in all tested dimensions. This may be explained by the increased correlation length of function $f_{10}$ as compared to $f_9$ Fig. 3.8.
Robustness against noise can be probed on the pair $f_2/f_4$. For $n = 10$, success rate and performance of PS-CMA-ES are similar. In higher dimensions, however, PS-CMA-ES is not able to reach the required accuracy for solving $f_4$. We infer that, for unimodal topologies, noise affects PS-CMA-ES more than it does for IPOP-CMA-ES.

| n=10 | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Func.** | **min** | **median** | **max** | **mean** | **std** | **$p_s$** | **Succ. Perf.** |
| f1 | 2.13e+04 | 2.34e+04 | 2.44e+04 | 2.33e+04 | 8.39e+02 | 1.00 | 2.33e+04 |
| f2 | 3.20e+04 | 3.56e+04 | 3.75e+04 | 3.56e+04 | 1.26e+03 | 1.00 | 3.56e+04 |
| f3 | - | - | - | - | - | 0.00 | - |
| f4 | 3.54e+04 | 3.91e+04 | 4.10e+04 | 3.86e+04 | 1.72e+03 | 1.00 | 3.86e+04 |
| f5 | 7.38e+04 | 7.99e+04 | 8.28e+04 | 7.93e+04 | 2.49e+03 | 1.00 | 7.93e+04 |
| f6 | 4.47e+04 | 8.60e+04 | 9.49e+04 | 8.20e+04 | 1.29e+04 | 1.00 | 8.20e+04 |
| f7 | 2.10e+04 | 2.42e+04 | 2.63e+04 | 2.41e+04 | 1.38e+03 | 1.00 | 2.41e+04 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 4.48e+04 | 7.81e+04 | 8.07e+04 | 7.04e+04 | 1.38e+04 | 1.00 | 7.04e+04 |
| f10 | 4.66e+04 | 7.87e+04 | - | 7.35e+04 | 1.17e+04 | 0.96 | 7.66e+04 |
| f11 | 4.05e+04 | - | - | 4.57e+04 | 3.77e+03 | 0.16 | 2.86e+05 |
| f12 | 2.98e+04 | 6.08e+04 | - | 5.61e+04 | 1.81e+04 | 0.72 | 7.79e+04 |
| f13 | - | - | - | - | - | 0.00 | - |
| f14 | - | - | - | - | - | 0.00 | - |
| f15 | 9.17e+04 | - | - | 9.17e+04 | 0.00e+00 | 0.04 | 2.29e+06 |
| n=30 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | **$p_s$** | **Succ. Perf.** |
| f1 | 6.41e+04 | 6.64e+04 | 6.95e+04 | 6.65e+04 | 1.26e+03 | 1.00 | 6.65e+04 |
| f2 | 2.43e+05 | 2.55e+05 | 2.63e+05 | 2.55e+05 | 5.65e+03 | 1.00 | 2.55e+05 |
| f3 | - | - | - | - | - | 0.00 | - |
| f4 | - | - | - | - | - | 0.00 | - |
| f5 | 2.96e+05 | - | - | 2.96e+05 | 0.00e+00 | 0.04 | 7.39e+06 |
| f6 | - | - | - | - | - | 0.00 | - |
| f7 | 6.75e+04 | 7.35e+04 | 7.76e+04 | 7.33e+04 | 2.05e+03 | 1.00 | 7.33e+04 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 2.63e+05 | - | - | 2.63e+05 | 0.00e+00 | 0.04 | 6.56e+06 |
| f10 | 2.62e+05 | - | - | 2.62e+05 | 0.00e+00 | 0.04 | 6.55e+06 |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | - | - | - | - | - | 0.00 | - |
| n=50 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | **$p_s$** | **Succ. Perf.** |
| f1 | 1.00e+05 | 1.03e+05 | 1.05e+05 | 1.03e+05 | 1.08e+03 | 1.00 | 1.03e+05 |
| f2 | - | - | - | - | - | 0.00 | - |
| f3 | - | - | - | - | - | 0.00 | - |
| f4 | - | - | - | - | - | 0.00 | - |
| f5 | - | - | - | - | - | 0.00 | - |
| f6 | - | - | - | - | - | 0.00 | - |
| f7 | 1.09E+05 | 1.13E+05 | 1.32E+05 | 1.15E+05 | 5.82E+03 | 1.00 | 1.15E+05 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | - | - | - | - | - | 0.00 | - |
| f10 | - | - | - | - | - | 0.00 | - |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | - | - | - | - | - | 0.00 | - |

Table 4.5: Number of FES (min, median, maximum, mean, and standard deviation) for PS-CMA-ES **without** LD sampling to reach the required accuracy for $f_1$-$f_{15}$ in $n = 10$ and $f_1$–$f_{12}$ in $n = 30, 50$ within MAX_FES $= 10^4 n$. Columns 7 and 8 give the empirical success rate $p_s$ and the success performance.

| n=10 | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Func.** | **min** | **median** | **max** | **mean** | **std** | **p$_s$** | **Succ. Perf.** |
| f1 | 1.89e+04 | 1.97e+04 | 2.07e+04 | 1.98e+04 | 4.05e+02 | 1.00 | 1.98e+04 |
| f2 | 2.72e+04 | 2.88e+04 | 3.00e+04 | 2.87e+04 | 8.27e+02 | 1.00 | 2.87e+04 |
| f3 | 9.28e+04 | - | - | 9.83e+04 | 3.29e+03 | 0.32 | 3.07e+05 |
| f4 | 2.80e+04 | 3.01e+04 | 3.16e+04 | 3.02e+04 | 1.02e+03 | 1.00 | 3.02e+04 |
| f5 | 6.50e+04 | 6.64e+04 | 6.88e+04 | 6.68e+04 | 9.52e+02 | 1.00 | 6.68e+04 |
| f6 | 4.63e+04 | 7.54e+04 | 8.39e+04 | 7.28e+04 | 9.72e+03 | 1.00 | 7.28e+04 |
| f7 | 1.86e+04 | 1.92e+04 | 2.10e+04 | 1.97e+04 | 7.56e+02 | 1.00 | 1.97e+04 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 1.48e+04 | 7.41e+04 | 8.02e+04 | 6.30e+04 | 1.64e+04 | 1.00 | 6.30e+04 |
| f10 | 4.82e+04 | 7.67e+04 | 7.92e+04 | 6.75e+04 | 1.36e+04 | 1.00 | 6.75e+04 |
| f11 | 3.00e+04 | - | - | 3.08e+04 | 7.39e+02 | 0.12 | 2.57e+05 |
| f12 | 2.44e+04 | 3.91e+04 | 9.94e+04 | 4.81e+04 | 2.46e+04 | 1.00 | 4.81e+04 |
| f13 | - | - | - | - | - | 0.00 | - |
| f14 | - | - | - | - | - | 0.00 | - |
| f15 | 9.15e+04 | - | - | 9.35e+04 | 2.20e+03 | 0.12 | 7.79e+05 |
| n=30 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 5.13e+04 | 5.36e+04 | 5.38e+04 | 5.36e+04 | 6.62e+02 | 1.00 | 5.36e+04 |
| f2 | 1.94e+05 | 2.05e+05 | 2.06e+05 | 2.04e+05 | 3.73e+03 | 1.00 | 2.04e+05 |
| f3 | - | - | - | - | - | 0.00 | - |
| f4 | - | - | - | - | - | 0.00 | - |
| f5 | - | - | - | - | - | 0.00 | - |
| f6 | - | - | - | - | - | 0.00 | - |
| f7 | 5.49e+04 | 5.87e+04 | 6.46e+04 | 5.91e+04 | 2.02e+03 | 1.00 | 5.91e+04 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 1.70e+05 | - | - | 2.17e+05 | 4.87e+04 | 0.16 | 1.36e+06 |
| f10 | 1.65e+05 | 2.51e+05 | - | 2.15e+05 | 2.96e+04 | 0.64 | 3.36e+05 |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | 2.28e+05 | - | - | 2.28e+05 | 0.00e+00 | 0.04 | 5.70e+06 |
| n=50 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 8.36e+04 | 8.45e+04 | 8.64e+04 | 8.49e+04 | 7.55e+02 | 1.00 | 8.49e+04 |
| f2 | 5.00e+05 | 5.00e+05 | 5.00e+05 | 5.00e+05 | 0.00e+00 | 1.00 | 5.00e+05 |
| f3 | - | - | - | - | - | 0.00 | - |
| f4 | - | - | - | - | - | 0.00 | - |
| f5 | - | - | - | - | - | 0.00 | - |
| f6 | - | - | - | - | - | 0.00 | - |
| f7 | 8.81e+04 | 9.48e+04 | 9.70e+04 | 9.43e+04 | 2.23e+03 | 1.00 | 9.43e+04 |
| f8 | - | - | - | - | - | 0.00 | - |
| f9 | 3.34e+05 | - | - | 3.34e+05 | 0.00e+00 | 0.04 | 8.35e+06 |
| f10 | 3.34e+05 | - | - | 3.58e+05 | 3.36e+04 | 0.08 | 4.47e+06 |
| f11 | - | - | - | - | - | 0.00 | - |
| f12 | - | - | - | - | - | 0.00 | - |

Table 4.6: Number of FES (min, median, maximum, mean, and standard deviation) for PS-CMA-ES **with** LD sampling to reach the required accuracy for $f_1$-$f_{15}$ in $n = 10$ and $f_1$–$f_{12}$ in $n = 30, 50$ within MAX_FES $= 10^4 n$. Columns 7 and 8 give the empirical success rate $p_s$ and the success performance.

**PS-CMA-ES performance on multi-modal/multi-funnel landscapes.** We analyze the performance of PS-CMA-ES (with LD sampling) on the subset $f_8$–$f_{25}$ that only includes hard multi-modal/multi-funnel functions with exponentially many minima. The majority of these functions cannot be solved within the allowed FES budget. We thus use the mean (over all 25 repetitions of each problem) best ever found function value error after MAX_FES=$10^4 n$ to rank the different methods for performance comparison. The summary statistics are given in Table 4.7 for $n = 10$, Table 4.8 for $n = 30$, and Table 4.9 for $n$=50.

Compared to LR-CMA-ES, IPOP-CMA-ES, and PSGES ($n$=10 only), PS-CMA-ES achieves the best average rank over all functions and all dimensions. In $n = 10$, PS-CMA-ES is always ranked first or second, except for $f_{22}$ where it is outperformed by IPOP- and LR-CMA-ES. The average rank is 1.77. In $n = 30$, PS-CMA-ES is ranked first or second, except for $f_{18}$, $f_{20}$ and $f_{23}$. In $n = 50$, PS-CMA-ES is ranked first or second, except for $f_{19}$. The average rank for $n = 30, 50$ is 1.5. PS-CMA-ES shows remarkable performance on the triplet $f_{15}$–$f_{17}$. Function $f_{16}$ is the rotated version of $f_{15}$, and $f_{17}$ is a noisy version of $f_{16}$, suggesting that PS-CM-ES is robust against rotation and addition of noise on multi-funnel landscapes. In some cases, the mean error is orders of magnitudes lower than that of IPOP-CMA-ES.

| Func. | PS-CMA-ES | LR-CMA-ES | IPOP-CMA-ES | PSGES |
|---|---|---|---|---|
| f8 | 2.00e+01 (1) | 2.00e+01 (1) | 2.00e+01 (1) | 2.09e+01 (4) |
| f9 | 3.98e-02 (1) | 4.49e+01 (4) | 2.39e-01 (2) | 3.46e+00 (3) |
| f10 | 7.73e-09 (1) | 4.08e+01 (4) | 7.96e-02 (2) | 1.46e+01 (3) |
| **f11** | **8.54e-01 (1)** | **3.65e+00 (3)** | **9.34e-01 (2)** | **1.35e+01 (4)** |
| **f12** | **1.10e+00 (1)** | **2.09e+02 (3)** | **2.93e+01 (2)** | **3.60e+02 (4)** |
| **f13** | **3.67e-01 (1)** | **4.94e-01 (2)** | **6.96e-01 (3)** | **8.21e-01 (4)** |
| f14 | 3.40e+00 (2) | 4.01e+00 (3) | 3.01e+00 (1) | 5.00e+00 (4) |
| **f15** | **8.67e+01 (1)** | **2.11e+02 (2)** | **2.28e+02 (3)** | **3.26e+02 (4)** |
| **f16** | **9.28e+01 (1)** | **1.05e+02 (2)** | **9.31e+04 (4)** | **2.01e+02 (3)** |
| **f17** | **1.12e+02 (1)** | **5.49e+02 (3)** | **1.23e+02 (2)** | **3.03e+03 (4)** |
| **f18** | **3.60e+02 (2)** | **4.97e+02 (3)** | **3.32e+02 (1)** | **7.15e+02 (4)** |
| **f19** | **3.25e+02 (1)** | **5.16e+02 (3)** | **3.26e+02 (2)** | **6.69e+02 (4)** |
| **f20** | **3.43e+02 (2)** | **4.42e+02 (3)** | **3.00e+02 (1)** | **7.05e+02 (4)** |
| **f21** | **4.71e+02 (2)** | **4.04e+02 (1)** | **5.00e+02 (3)** | **8.89e+02 (4)** |
| **f22** | **7.46e+02 (3)** | **7.04e+02 (1)** | **7.29e+02 (2)** | **8.11e+02 (4)** |
| **f23** | **5.58e+02 (2)** | **7.91e+02 (3)** | **5.59e+02 (1)** | **1.08e+03 (4)** |
| **f24** | **2.00e+02 (1)** | **8.65e+02 (4)** | **2.00e+02 (1)** | **4.19e+02 (3)** |
| **f25** | **4.03e+02 (2)** | **4.42e+02 (4)** | **3.74e+02 (1)** | **4.15e+02 (3)** |
| Rank | 1.77 (32/18) | 2.72 (49/18) | 1.88 (34/18) | 3.5 (63/18) |

Table 4.7: Mean function error and relative ranks (in brackets) after $10^5$ function evaluations in $n = 10$ dimensions for PS-CMA-ES with LD sampling, IPOP-CMA-ES, LR-CMA-ES and PSGES. Multi-funnel landscapes are in bold. The last row shows the average rank over all functions.

In summary, we conclude that PS-CMA-ES with LD sampling is a good choice on arbitrary landscape topologies for problems of low dimensionality ($n \approx 10$). In higher dimensions, PS-CMA-ES is a good choice for highly multi-modal single-funnel, multi-funnel, and noisy problems. This is further supported by numerical experiments on Lunacek's double-funnel

| Func. | PS-CMA-ES | LR-CMA-ES | IPOP-CMA-ES |
|---|---|---|---|
| f8 | 2.00e+01 (1) | 2.00e+01 (1) | 2.01e+01 (3) |
| f9 | 8.76e-01 (1) | 2.91e+02 (3) | 9.38e-01 (2) |
| f10 | 5.57e-01 (1) | 5.63e+02 (3) | 1.65e+00 (2) |
| **f11** | **7.10e+00 (2)** | **1.52e+01** (3) | **5.48e+00 (1)** |
| **f12** | **8.80e+02 (1)** | **1.32e+04** (2) | **4.43e+04 (3)** |
| **f13** | **2.05e+00 (1)** | **2.32e+00 (2)** | **2.49e+00 (3)** |
| f14 | 1.24e+01 (1) | 1.40e+01 (3) | 1.29e+01 (2) |
| **f15** | **1.37e+02 (1)** | **2.16e+02** (3) | **2.08e+02 (2)** |
| **f16** | **1.59e+01 (1)** | **5.84e+01** (3) | **3.50e+01 (2)** |
| **f17** | **9.15e+01 (1)** | **1.07e+03** (3) | **2.91e+02 (2)** |
| **f18** | **9.05e+02 (3)** | **8.90e+02** (1) | **9.04e+02 (2)** |
| **f19** | **8.85e+02 (1)** | **9.03e+02** (2) | **9.04e+02 (3)** |
| **f20** | **9.05e+02 (3)** | **8.89e+02** (1) | **9.04e+02 (2)** |
| **f21** | **5.00e+02 (2)** | **4.85e+02** (1) | **5.00e+02 (2)** |
| **f22** | **8.43e+02 (2)** | **8.71e+02** (3) | **8.03e+02 (1)** |
| **f23** | **5.43e+02 (3)** | **5.35e+02** (2) | **5.34e+02 (1)** |
| **f24** | **2.10e+02 (1)** | **1.41e+03** (3) | **9.10e+02 (2)** |
| **f25** | **2.10e+02 (1)** | **6.91e+02** (3) | **2.11e+02 (2)** |
| Rank | 1.5 (27/18) | 2.33 (42/18) | 2.05(37/18) |

Table 4.8: Mean function error and relative ranks (in brackets) after $3 \cdot 10^5$ function evaluations in $n = 30$ dimensions for PS-CMA-ES with LD sampling, IPOP-CMA-ES, and LR-CMA-ES. Multi-funnel landscapes are in bold. The last row shows the average rank over all functions.

| Func. | PS-CMA-ES | LR-CMA-ES | IPOP-CMA-ES |
|---|---|---|---|
| f8 | 2.00e+01 (1) | 2.00e+01 (1) | 2.01e+01 (3) |
| f9 | 5.45e+00 (2) | 5.67e+02 (3) | 1.39e+00 (1) |
| f10 | 5.33e+00 (2) | 1.48e+03 (3) | 1.72e+00 (1) |
| **f11** | **1.59e+01 (2)** | **3.41e+01 (3)** | **1.17e+01 (1)** |
| **f12** | **6.90e+03 (1)** | **8.93e+04 (2)** | **2.27e+05 (3)** |
| **f13** | **4.15e+00 (1)** | **4.70e+00 (3)** | **4.59e+00 (2)** |
| f14 | 2.15e+01 (1) | 2.39e+01 (3) | 2.29e+01 (2) |
| **f15** | **1.25e+02 (1)** | **2.50e+02 (3)** | **2.04e+02 (2)** |
| **f16** | **1.62e+01 (1)** | **7.09e+01 (3)** | **3.09e+01 (2)** |
| **f17** | **9.13e+01 (1)** | **1.05e+03 (3)** | **2.34e+02 (2)** |
| **f18** | **8.70e+02 (1)** | **9.06e+02 (2)** | **9.13e+02 (3)** |
| **f19** | **9.13e+02 (3)** | **9.11e+02 (1)** | **9.12e+02 (2)** |
| **f20** | **9.09e+02 (2)** | **9.01e+02 (1)** | **9.12e+02 (3)** |
| **f21** | **6.62e+02 (2)** | **5.00e+02 (1)** | **1.00e+03 (3)** |
| **f22** | **8.63e+02 (2)** | **9.10e+02 (3)** | **8.05e+02 (1)** |
| **f23** | **8.12e+02 (2)** | **6.37e+02 (1)** | **1.01e+03 (3)** |
| **f24** | **2.00e+02 (1)** | **8.43e+02 (2)** | **9.55e+02 (3)** |
| **f25** | **2.14e+02 (1)** | **4.77e+02 (3)** | **2.15e+02 (2)** |
| Rank | 1.5 (27/18) | 2.23 (41/18) | 2.17 (39/18) |

Table 4.9: Mean function error and relative ranks (in brackets) after $5 \cdot 10^5$ function evaluations in $n = 50$ dimensions for PS-CMA-ES with LD sampling, IPOP-CMA-ES, and LR-CMA-ES. Multi-funnel landscapes are in bold. The last row shows the average rank over all functions.

landscapes (see Eqs. (2.11) and 2.12). We refer to the Master Theses (Ofenbeck, 2009; König, 2010) for these results.

### 4.3.3 Conclusions and future work

We have introduced the concept of parallel CMA-ES as a general extension to CMA-ES. One instance of such a scheme, the Particle Swarm CMA Evolution Strategy, has been presented in detail. There, each CMA-ES instance is considered an individual swarm particle. Global knowledge is included in the CMA-ES sampling distribution using two mechanisms: (1) rotation of the covariance matrix such that the longest eigenvector points in the direction of the globally best, and (2) biasing the mean of the sampling distribution toward the global best. We have described a computationally efficient algorithm for uniquely rotating the covariance matrix in high-dimensional spaces. The presented method adds five strategy parameters: the swarm size $S$, the mixing weight $c_{\mathrm{p}}$, the biasing parameters $t_{\mathrm{c}}$ and $b$, and the communication interval $I_{\mathrm{c}}$. The biasing rules for the sampling mean provide additional means of accounting for prior knowledge about the optimization problem at hand. We have determined, in an unbiased way, standard values for all strategy parameters that provide good average performance on the wide range of test functions represented in the CEC 2005 benchmark suite, hence rendering PS-CMA-ES practically parameter free. Using these standard parameters, we have evaluated the performance of PS-CMA-ES and compared it to LR-CMA-ES, the IPOP-CMA-ES, and PSGES, another hybrid particle swarm evolution strategy (Hsieh et al., 2007).

Our benchmarks have shown the superior performance of PS-CMA-ES on strongly multi-modal problems (Rastrigin function, $f_9/f_{10}$), and multi-funnel problems across all tested dimensions. We believe that PS-CMA-ES benefits from an increased global exploration power introduced by the swarm communication. Analyzing the search space coverage of the different algorithms will be a topic of future research.

With standard parameter settings, PS-CMA-ES presented no advantage over the reference algorithms on unimodal functions and several basic multi-modal functions. This could potentially be improved by tuning the strategy parameters of PS-CMA-ES specifically for these cases, as indicated by our parameter sweep for $n = 10$ dimensions. We observed that runs without global communication and smaller swarm sizes would perform better on these problems. This was expected since the standard CMA-ES approximates well the fitness landscape of unimodal functions and communication can only disturb its convergence. For larger swarm sizes, there is only little time for individual CMA-ES instances to converge within the maximum allowed FES budget.

While PS-CMA-ES performs well using the standard parameter settings determined in this work, its performance could be further improved by refined parameter choices. In particular, it could be advantageous to couple the communication interval and the swarm size to the problem dimension. Alternative communication topologies are also conceivable. For instance, ring, toroidal or grid topologies could be considered instead of the all-to-all topology used here. Preliminary results are already available in (König, 2010).

PS-CMA-ES comes at an increased computational cost compared to restart CMA-ES strategies. The main computational overhead is caused by the rotation of the covariance matrix. The presented rotation algorithm requires $2n\,(n-1)+1$ matrix multiplications at each PSO update and for each swarm member in order to construct $\mathbf{R}_\mathrm{p}$ and $\mathbf{R}_\mathrm{b}$. Hence, the computational cost increases quadratically with the number of dimensions and linearly with swarm size. This can, to a certain extent, be relaxed by choosing the smallest possible swarm size and performing PSO updates less frequently (controlled through the strategy parameter $I_\mathrm{c}$). The overall scaling is dictated by the final matrix multiplication to construct $\mathbf{R}$, which scales at most cubically with dimension. In addition, alternative biasing schemes that do not require $n$-dimensional rotations could be investigated, for instance, by using the direction toward the current global best solution for a weighted rank-one update of the individual covariance matrices. It is also noteworthy that the intrinsic parallelism of parallel CMA-ES schemes allows leveraging the computational performance of multi-core platforms, in particular when the swarm size is chosen as an integer multiple of the number of processing cores. PS-CMA-ES thus provides a straightforward way to benefit from the anticipated future increase in the number of cores per chip by using this parallelism to increase the exploration power of the search. This was also the driving motivation for the development of the parallel software library pCMALib. In pCMALib, individual CMA-ES instances are distinct processes that run independently on different cores. Communication is realized using the Message Passing Interface (MPI). The design and implementation of pCMALib, as well as all available settings for parallel CMA-ES schemes are summarized in Appendix A2.

## 4.4 Gaussian Adaptation

Despite its sound theoretical design principles and its overwhelming practical success, CMA-ES is often confronted with the criticism that its internal update rules are rather complex, and its parameter settings are derived in an *ad hoc* fashion (Beyer and Sendhoff, 2008). A number of recent publications attempt to alleviate these drawbacks. Beyer and Sendhoff introduced the Covariance Matrix Self Adaptation ES (CMSA-ES) (Beyer and Sendhoff, 2008), comprising simpler rules for the covariance matrix and step size adaptation, and fewer parameters. In a series of papers, Natural Evolution Strategies (NES) (Wierstra et al., 2008; Sun et al., 2009; Glasmachers et al., 2010) have been proposed as a more "principled" approach to black-box optimization. NES is designed to follow the *Natural Gradient* of the landscape, a well-known concept for Machine Learning (Amari, 1998). Akimoto and co-workers recently made the link between NES and CMA-ES explicit and showed a bi-directional relationship between CMA-ES and NES (Akimoto et al., 2011). Here, we offer a different view on covariance matrix and step size adaptation that is rooted in the Maximum Entropy principle: *Gaussian Adaptation*.

### 4.4.1 Gaussian Adaptation and the Maximum Entropy Principle

Gaussian Adaptation (GaA) has been developed in the context of electrical network design. There, the key goal is to find an optimal setting of design parameters $\mathbf{x} \in \mathbb{R}^n$, e.g., nominal values of resistances and capacities in an analog network, that fulfill two requirements: First, the parameter settings satisfy the specifications imposed by the engineer, i.e. some

(real-valued) objective (or criterion) function $f(\mathbf{x})$ applied to the network output, and second, the nominal values should be robust with respect to intrinsic random variations of the components during operation of the electrical device. In the late 1960's Gregor Kjellström, an engineer at Ericsson, realized that with increasing network complexity classical optimizers such as conjugate gradients perform poorly, especially when analytical gradients are not readily available or when the objective function is multi-modal. He suggested to search the space of valid parameter settings with stochastic methods that only rely on evaluations of the objective function. Starting from an exploration method that can be considered an adaptive random walk through design space (Kjellström, 1969, 1970), he refined his algorithm to what he called Gaussian Adaptation (Kjellström and Taxen, 1981).

Before turning to the problem of optimization, Kjellström considered the following simpler situation: Assume that the engineer of an electrical circuit can vary the set of design parameters and can decide whether these settings fulfill a specified criterion or not. How can one describe the set $\mathcal{A} \subset \mathbb{R}^n$ of acceptable solutions in a general and compact manner? Based on Shannon's information theory, Kjellström derived that under the assumption of finite mean $\mathbf{m}$ and covariance $\mathbf{C}$ of the samples, a Gaussian distribution may be used to characterize $\mathcal{A}$ (Kjellström and Taxen, 1981). Although not specifically stated in the original publication, Kjellström applied the maximum entropy principle, developed by Jaynes in 1957 (Jaynes, 1957). There, Jaynes states that the Maximum Entropy principle "is the least biased estimate possible on the given information; i.e., it is maximally noncommittal with regard to missing information." In the case of given mean and covariance information, the Gaussian distribution maximizes the entropy $\mathcal{H}$, and hence is the preferred choice to describe the region of acceptable points. The entropy of a multivariate Gaussian distribution is:

$$\mathcal{H}(\mathcal{N}) = \log\left(\sqrt{(2\pi e)^n \det(\mathbf{C})}\right) , \tag{4.21}$$

where $\mathbf{C}$ is the covariance matrix. In order to get the most informative characterization of the region $\mathcal{A}$, Kjellström envisioned an iterative sampling strategy with a Gaussian distribution that satisfies the following criteria: (i) The probability of finding a feasible design parameter set should be fixed to a predefined value $P < 1$, and (ii) the spread of the samples quantified by their entropy should be maximized. As Eq. (4.21) shows, this can be achieved by maximizing the determinant of the covariance matrix. In the situation where the parameters have to fulfill a predefined static criterion, the iterative sampler should push the mean of the distribution toward the *center* of the feasible design space. Simultaneously, it should adapt the orientation and scale of the covariance matrix to the shape of $\mathcal{A}$ under the constraint of the fixed hitting probability. The final mean can, e.g., be used as the nominal design parameter set. Figure 4.7 illustrates this process, which is called "design centering" or "design tolerancing" in electrical engineering (Graeb, 2009). When the criterion function $f(\mathbf{x})$ yields real values, the sampler can be turned into a minimizer by introducing a fitness acceptance threshold $c_\mathrm{T}$. For a given threshold, GaA attempts to adapt a Gaussian distribution to the largest landscape region where Gaussian sample points are below the threshold with probability $P$ . In the course of minimization, $c_\mathrm{T}$ is monotonically lowered until some convergence criteria are met (Kjellström and Taxen, 1981). It is noteworthy that the idea of threshold acceptance has later been reintroduced by Dueck and Scheuer in the Threshold Acceptance (TA) algorithm (Dueck and Scheuer, 1990), an extension of Simulated Annealing (SA) algorithm (Kirkpatrick et al., 1983).

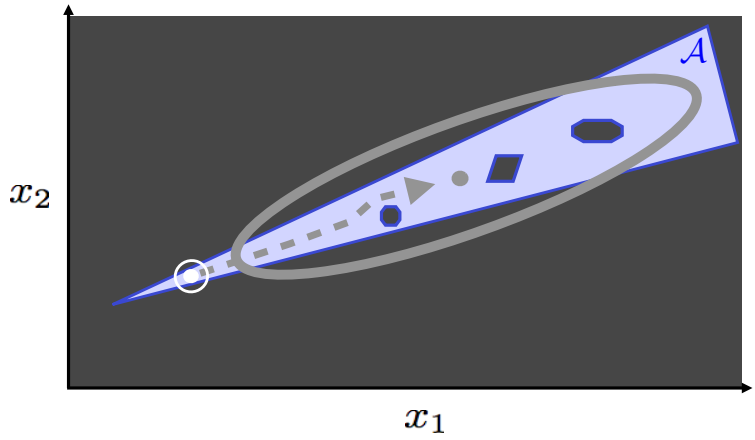



Figure 4.7: Illustration of Gaussian Adaptation. The light purple, non-convex area defines the acceptable region $\mathcal{A}$ in a 2D design-parameter space $\mathbf{x}$. Both the left (white) and right (gray) dots and ellipsoids represent the means and covariances of two Gaussian distributions with the same hitting probability $P$. GaA moves away from the left corner toward the center and adapts the distribution to the shape of $\mathcal{A}$.

### 4.4.2 The Gaussian Adaptation algorithm

In order to realize an iterative procedure that works for both design tolerancing and optimization, Kjellström proposed the Gaussian Adaptation algorithm. The process starts by setting the mean $\mathbf{m}^{(0)}$ of a multivariate Gaussian to an initial point $\mathbf{x}^{(\mathbf{0})} \in \mathcal{A}$. The covariance $\mathbf{C}^{(g)}$ is decomposed as follows:

$$\mathbf{C}^{(g)} = \left(r \cdot \mathbf{Q}^{(g)}\right)\left(r \cdot \mathbf{Q}^{(g)}\right)^T = r^2 \left(\mathbf{Q}^{(g)}\right)\left(\mathbf{Q}^{(g)}\right)^T , \tag{4.22}$$

where $r$ is the scalar step size and $\mathbf{Q}^{(g)}$ is the normalized square root of $\mathbf{C}^{(g)}$. Like in CMA-ES, $\mathbf{Q}^{(g)}$ is found by eigendecomposition of the covariance matrix $\mathbf{C}^{(g)}$. The initial $\mathbf{Q}^{(0)}$ is set to the identity matrix $\mathbf{I}$. In iteration $g+1$ a single point is sampled from a Gaussian distribution according to:

$$\mathbf{x}^{(g+1)} = \mathbf{m}^{(g)} + r^{(g)}\mathbf{Q}^{(g)}\eta^{(g)} , \tag{4.23}$$

where $\eta^{(g)} \sim \mathcal{N}\Big(\mathbf{0}, \mathbf{I}\Big)$. The new sample is evaluated by the criterion function $f(\mathbf{x}^{(\mathbf{g+1})})$. *Only if* the sample fulfills the specification, i.e. $\mathbf{x}^{(g+1)} \in \mathcal{A}$ in the design-tolerancing scenario or $f(\mathbf{x}^{(g+1)}) < c_{\mathrm{T}}^{(g)}$ in the optimization scenario, the following adaptation rules are applied: The step size $r$ is increased according to $r^{(g+1)} = f_{\mathrm{e}} \cdot r^{(g)}$, where $f_{\mathrm{e}} > 1$ is called the expansion factor. The mean is updated via

$$\mathbf{m}^{(g+1)} = \left(1 - \frac{1}{N_{\mathrm{m}}}\right)\mathbf{m}^{(g)} + \frac{1}{N_{\mathrm{m}}}\mathbf{x}^{(g+1)} . \tag{4.24}$$

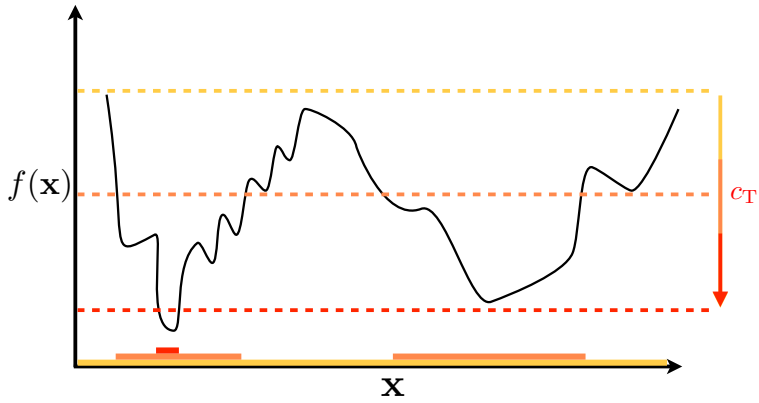



Figure 4.8: Illustration of lowering the fitness threshold $c_{\mathrm{T}}$ in GaA. The dashed lines represent a certain value of $c_{\mathrm{T}}$. The bars on the x-axis show the corresponding acceptable regions of the landscape.

$N_{\mathrm{m}}$ is a weighting factor that controls how fast the mean is shifted. The covariance matrix is updated through:

$$\mathbf{C}^{(g+1)} = \left(1 - \frac{1}{N_{\mathrm{C}}}\right)\mathbf{C}^{(g)} + \frac{1}{N_{\mathrm{C}}}\left(\mathbf{x}^{(g+1)} - \mathbf{m}^{(g)}\right)\left(\mathbf{x}^{(g+1)} - \mathbf{m}^{(g)}\right)^{T}. \tag{4.25}$$

$N_{\mathrm{C}}$ weights the influence of the accepted sample point on the covariance matrix. Kjellström introduced an alternative update rule that is mathematically equivalent to Eq. 4.25, but numerically more robust. It acts directly on the square root $\mathbf{Q}^{(g)}$ of the covariance matrix:

$$\mathbf{\Delta C}^{(g+1)} = \left(1 - \frac{1}{N_{\mathrm{C}}}\right)\mathbf{I}^{(g)} + \frac{1}{N_{\mathrm{C}}}(\eta^{(g)})(\eta^{(g)})^{T}, \quad \mathbf{\Delta Q}^{(g+1)} = (\mathbf{\Delta C}^{(g+1)})^{\frac{1}{2}}. \tag{4.26}$$

$\mathbf{Q}^{(g+1)}$ is then computed as $\mathbf{Q}^{(g+1)} = \mathbf{Q}^{(g)}\mathbf{\Delta Q}^{(g+1)}$. In order to decouple the volume of the covariance (controlled by $r^{(g+1)}$) and its orientation, $\mathbf{Q}^{(g+1)}$ is normalized such that $\det(\mathbf{Q}^{(g+1)}) = 1$. As in CMA-ES, the full adaptation of the covariance matrix gives GaA the appealing property of being invariant to arbitrary rotations of the problem.

In case $\mathbf{x}^{(g+1)}$ is not accepted, only the step size is adapted according to $r^{(g+1)} = f_{\mathrm{c}} \cdot r^{(g)}$, where $f_{\mathrm{c}} < 1$ is the contraction factor.

A crucial ingredient for optimization using GaA is the dynamic lowering of the acceptance threshold $c_{\mathrm{T}}$. A natural choice would be to set-up a certain landscape-independent schedule that monotonically decreases $c_{\mathrm{T}}$, reminiscent of the various cooling schedules developed for Simulated Annealing. Kjellström, however, suggested to use the following adaptation rule:

$$\mathbf{c}_{\mathrm{T}}{}^{(g+1)} = \left(1 - \frac{1}{N_{\mathrm{T}}}\right)\mathbf{c}_{\mathrm{T}}{}^{(g)} + \frac{1}{N_{\mathrm{T}}} f(\mathbf{x}^{(g+1)}), \tag{4.27}$$

where $N_\mathrm{T}$ controls the weighting between the old threshold and the objective value of the *accepted* sample. It can readily be seen that this fitness-dependent threshold update leaves the algorithm invariant to linear transformations of the objective function but not general monotonic transformations as in CMA-ES.

**Step size adaptation in GaA.** GaA offers a genuinely different view on step size control than CMA-ES or ASSRS. In CMA-ES, the step size is adapted by monitoring the correlation between subsequent steps in the evolution path. In ASSRS (and the (1+1)-ES), the step size is adapted in order to yield optimal progress on quadratic functions. In GaA, we maximize the entropy of the search distribution, which suggests increasing the step size as soon as we have found an acceptable solution and decreasing it otherwise. We first consider the hitting (acceptance) probability $P$. Kjellström investigated the information-theoretic optimality of $P$ for a random walk in a simplex region (Kjellström, 1969) and for Gaussian Adaptation in general regions (Kjellström and Taxen, 1981). In both cases he concluded that the efficiency of the process and $P$ are related as $E \propto -P \log P$, leading to the optimal $P = \frac{1}{e} \approx 0.3679$, where $e$ is Euler's number. A proof is provided in (Kjellström, 1991). Maintaining this hitting probability corresponds to leaving the volume of the distribution, $\det(\mathbf{C})$, constant under stationary conditions. As $\det(\mathbf{C}) = r^{2n} \det(\mathbf{Q}\mathbf{Q}^T)$, the expansion and contraction factors $f_\mathrm{e}$ and $f_\mathrm{c}$ increase or decrease the volume by a factor of $f_\mathrm{e}^{2n}$ or $f_\mathrm{c}^{2n}$, respectively. After $S$ successful and $F$ failed samples, a necessary condition for constant volume thus is:

$$\prod_{i=1}^{S} (f_\mathrm{e})^{2n} \prod_{i=1}^{F} (f_\mathrm{c})^{2n} = 1\,. \tag{4.28}$$

Using $P = \frac{S}{S+F}$ and introducing a small $\beta > 0$, one can verify that $f_\mathrm{e} = 1 + \beta(1-P)$ and $f_\mathrm{c} = 1 - \beta P$ satisfies Eq. (4.28) to first order.

**Further parameter settings in GaA.** In the original articles, Kjellström does not provide standard settings for the strategy parameters, $\beta$, $N_\mathrm{m}$, $N_\mathrm{C}$, and $N_\mathrm{T}$. After personal communication with him and insight from the rank-one update in CMA-ES, we propose to couple the scalar rate $\beta$ to the strategy parameters $N_\mathrm{C}$ and $N_\mathrm{T}$, but not to $N_\mathrm{m}$. Since $N_\mathrm{m}$ influences the update of $\mathbf{m} \in \mathbb{R}^n$, it is reasonable to set $N_\mathrm{m} \propto n$. We propose $N_\mathrm{m} = en$ in the general case. A similar reasoning is employed for $N_\mathrm{C}$ and $N_\mathrm{T}$. $N_\mathrm{C}$ influences the update of $\mathbf{C} \in \mathbb{R}^{n \times n}$, which contains $n^2$ entries. Hence, $N_\mathrm{C}$ should be related to $n^2$. We suggest using $N_\mathrm{C} = \frac{(n+1)^2}{\log(n+1)}$ as a standard value, and coupling $N_\mathrm{T} = \frac{N_\mathrm{C}}{2}$ and $\beta = \frac{1}{N_\mathrm{C}}$. Note that the parameter $c_\mathrm{cov}$ in CMA-ES corresponds to $1/N_\mathrm{C}$ in GaA.

**Relation between GaA, ES and TA** There are several remarkable connections between GaA and classical ES. The canonical (1+1)-ES is, for example, a limit case of GaA. Setting $N_\mathrm{m} = N_\mathrm{T} = 1$ moves GaA's mean directly to the accepted sample and $c_\mathrm{T}$ to the fitness of the accepted sample. For $N_\mathrm{C} \to \infty$, the covariance remains isotropic and GaA becomes equivalent to the (1+1)-ES with a $P^\mathrm{th}$-success rule. Keeping $N_\mathrm{C}$ finite results in an algorithm that is almost equivalent to the (1+1)-CMA-ES (Igel et al., 2006). The (1+1)-CMA-ES is a single-sample variant of CMA-ES, where only a rank-one update of the covariance matrix is applied.

Four key differences to GaA, however, remain. First, the step size adaptation mechanism in (1+1)-CMA-ES uses the damped exponential function, allowing faster adaptation than in GaA (Igel et al., 2006). Second, (1+1)-CMA-ES uses information about the evolution path for the covariance matrix update, whereas GaA does not. Third, the decision of how to update the covariance is controlled by a threshold probability $p_{\text{thresh}}$ in (1+1)-CMA-ES. Only if the empirical acceptance probability $P_{\text{emp}}$ is below $p_{\text{thresh}}$, the current sample is used to update the evolution path. Finally, GaA normalizes the volume of the covariance matrix in order to decouple it from the step size; (1+1)-CMA-ES does not involve such a normalization.

When $N_{\text{C}} \to \infty$ and the threshold adaptation in Eq. (4.27) follows:

$$\mathbf{c}_{\text{T}}{}^{(g+1)} = f(\mathbf{x}^{(g)}) + T^{(g)}\,, \tag{4.29}$$

where $T^{(g)}$ is some monotonically decreasing sequence with limit 0 we arrive at Dueck's TA algorithm with fixed Gaussian proposal.

**Constraint handling and initialization** Kjellström does not provide any initialization or constraint handling techniques. We suggest that, in unconstrained optimization problems, GaA can be used as is. However, the starting point $\mathbf{m}^{(0)}$ and the initial step size $r^{(0)}$ must then be set by the user in a meaningful manner. In box-constrained optimization problems, boundaries are explicitly given by $\mathbf{x} \in [\mathbf{L}, \mathbf{U}] \subset \mathbb{R}^n$. Several boundary handling techniques can be employed. One can, e.g., reject points that fall outside the admissible hyper-rectangle, and resample. As in CMA-ES, this can become inefficient for search near the boundary. Especially in high dimensions, the probability of hitting the feasible region becomes small. It is also conceivable to employ boundary handling with quadratic penalty terms (Hansen and Kern, 2004), a method that has been successfully used in CMA-ES. In GaA, however, the boundary penalty would be problem specific, since GaA's search performance directly depends on the objective function values. We therefore suggest projecting the components of out-of-bounds samples onto the boundary along the coordinate axes, and evaluating the projected samples.

The initial mean $\mathbf{m}^{(0)}$ is drawn from a uniform distribution in the box $[\mathbf{L}, \mathbf{U}]$. The initial step size is set to $r^{(0)} = 1/e\,(\max \mathbf{U} - \min \mathbf{L})$, similar to the global search setting of the initial $\sigma$ in IPOP-CMA-ES (Auger and Hansen, 2005b). The initial threshold $c_{\text{T}}^{(0)}$ is set to $f(\mathbf{m}^{(0)})$.

### 4.4.3 Numerical examples

To illustrate the behavior of GaA with the developed strategy parameters, constraint handling, and initialization techniques, we provide numerical results on selected test functions. We show the convergence of GaA on quadratic functions by considering the sphere function. The covariance matrix adaptation mechanism is then demonstrated on Rosenbrock's function. Finally, we sketch the entropic behavior of GaA on the multi-modal function $\text{f}_{\text{Kjell}}$ as introduced by Kjellström (Kjellström and Taxen, 1992).

### *Gaussian Adaptation on the Sphere function*

We consider the sphere function (see Eq. (2.8)) as a prototypical quadratic function in order to study convergence of the GaA algorithm. For practical purposes, search is restricted to $\mathbf{x} \in [-5, 5]^n$. In order to study the dimension-dependence of GaA's convergence properties, we use the standard strategy parameter values, constraint handling, and initialization. 10 repetitions are conducted for dimensions $n = 2, 5, 10, 20, 30, 40, 50$. Figure 4.9 summarizes the results. We observe the expected log-linear convergence of GaA on the sphere function. Nev-

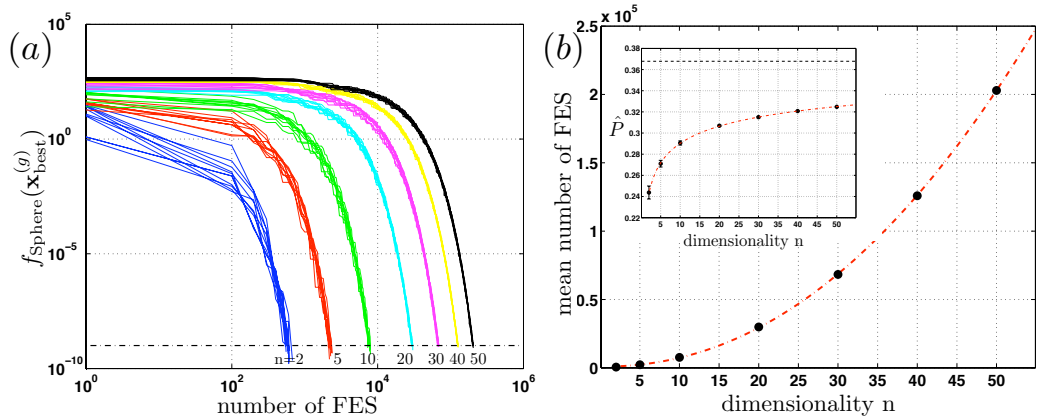



Figure 4.9: (a) Log-log plot of the current best fitness value $f(\mathbf{x}_{\text{best}}^{(g)})$ vs. the number of function evaluations (FES) on the sphere function for $n = 2, 5, 10, 20, 30, 40, 50$ (from left to right). The dashed line shows the target fitness (stopping criterion). (b) Average number of FES needed to reach the target fitness vs. dimensionality $n$. The dashed curve is a perfect fit of the power law $\text{FES}(n) = 47.11n^{2.138} + 857.8$. The inset shows the mean and standard deviation of the empirical hitting probability $\hat{P}$ vs. $n$. The dashed line represents the desired optimal $P = 1/e$.

ertheless, we show the results in a log-log plot in order to better discriminate the trajectories for different dimensions (Fig. 4.9a). The mean number of function evaluations (FES) needed to achieve an accuracy of $10^{-9}$ grows slightly faster than quadratically with $n$ (Fig. 4.9b). The measured (empirical) hitting probability $\hat{P}_{\text{emp}}$ approaches the optimal $P = 1/e$ with increasing dimension. A least-squares fit of a power law yields $\hat{P}(n) = -0.2077n^{-0.1831} + 0.4265$ (inset in Fig. 4.9b).

### *Gaussian Adaptation on Rosenbrock's function*

We study the behavior of GaA's covariance matrix adaptation on the multi-modal Rosenbrock's valley function (see Eq. (2.9)) . We perform 10 optimization runs with the same protocol for $n = 2, 5, 10, 20, 30, 40$. GaA finds the global minimum in all cases. Similar to CMA-ES, GaA's search on Rosenbrock can be divided into three phases: (1) log-linear convergence toward the origin; (2) a plateau region for covariance adjustment along the valley; (3) log-linear

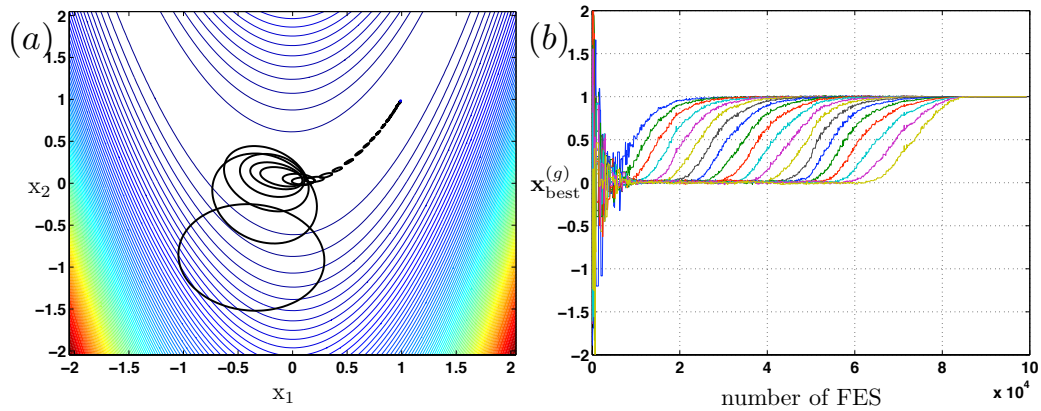



Figure 4.10: Typical trajectory of GaA on Rosenbrock's function for $n = 20$. (a) 2D Contour plot with the sub-covariances along the first two dimensions (black ellipses) shown every 1000 iterations. (b) Evolution of the components of $\mathbf{x}_{\text{best}}^{(g)}$ vs. the number of FES (= iterations) for the same run.

convergence near the global minimum. Fig. 4.10 shows a typical trajectory of GaA for $n = 20$, projected onto the first two dimensions. The same qualitative behavior is observed also in all other dimensions. After rapidly approaching the origin, GaA efficiently adapts its covariance to follow the valley. The objective variables then migrate, in order of increasing dimension, toward the global minimum. The mean number of function evaluations needed to achieve an accuracy of $10^{-9}$ follows the power law $\text{FES}(n) = 60.13n^{2.462} + 2807$ with larger offset, prefactor, and exponent being larger than on the sphere function. $\hat{P}$, however, converges toward the optimal value faster than on the sphere function: $\hat{P}(n) = -0.1405n^{-0.917} + 0.3582$ (see Fig. 4.11b).

### *Gaussian Adaptation on Kjellström's function*

We consider the highly multi-modal test function $f_{\text{Kjell}}$ as introduced by Kjellström (Kjellström and Taxen, 1992) (see also Eq. (2.13)). The standard setting for $N_{\text{C}}$ (and the coupled parameters $N_{\text{T}}$ and $\beta$) frequently results in premature convergence of the search into one of the $5^{25} \approx 3 \cdot 10^{17}$ local minima (success rate <10%). A simple parameter search on $N_{\text{C}}$ shows that the setting $N_{\text{C}} = 10n^2$ yields a better success rate on $f_{\text{Kjell}}$, namely 100% for up to $n = 50$. Figure 4.12b shows the trace of the mean $\mathbf{m}^{(g)}$ for a typical run in $n = 25$ dimensions with optimized $N_{\text{C}}$. In the first phase (dotted interval), GaA explores the entire search space (dotted interval in Fig. 4.12a). In the second phase (solid interval), it adjusts a high-entropy Gaussian distribution to the center of the broad region that contains low fitness values (solid interval in Fig. 4.12a). Finally, GaA proceeds to the region that contains the global minimum with function value $f_{\text{Kjell}}(\mathbf{x}_{\min}) \approx 0.9692^{25} \approx 0.4570$ (dashed interval).

Two conclusions can be drawn from these simulations: First, the function can efficiently be

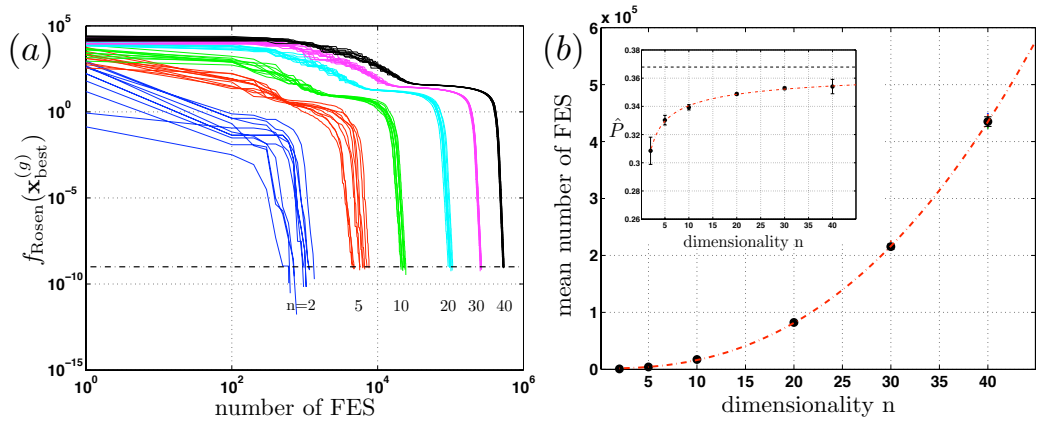



Figure 4.11: (a) Log-log plot of the current best fitness value $f(\mathbf{x}_{\text{best}}^{(g)})$ vs. the number of function evaluations (FES) on the Rosenbrock function for $n = 2, 5, 10, 20, 30, 40, 50$ (from left to right). The dashed line shows the target fitness (stopping criterion). (b) Average number of FES needed to reach the target fitness vs. dimensioniality $n$. The dashed curve is a perfect fit of the power law $\text{FES}(n) = 60.13n^{2.462} + 2807$. The inset shows the mean and standard deviation of the empirical hitting probability $\hat{P}$ vs. $n$. The dashed line represents the desired optimal $P = 1/e$.

solved by GaA despite the exponentially growing number of minima. This is due to the landscape topology, where the global minimum is located in a region that is amenable to hierarchical maximum-entropy adaptation. Second, we observe that the standard parameter settings are sub-optimal for this landscape. The problem of identifying optimal parameter settings for a given landscape topology can to some extent be alleviated by considering a restart strategy that iteratively adapts the strategy parameters.

### 4.4.4 Restart Gaussian Adaptation

Inspired by the success of adaptive restart strategies in CMA-ES, we also extend Gaussian Adaptation by such a mechanism. The resulting Restart GaA is a quasi parameter-free general-purpose black-box optimizer. The two ingredients of Restart GaA are the definition of suitable convergence criteria and the adaptation of internal strategy parameters in case of convergence to a local minimum.

**Convergence criteria.** In practical applications, it is unavoidable to define criteria that indicate convergence of GaA to a (local or global) minimum. We propose six convergence criteria: *MaxIter*, *TolFit*, *TolFun*, *TolX*, *TolR*, and *TolCon*.

1. *MaxIter*: GaA is stopped when a maximum number of allowed FES is reached. The default maximum budget is $10^4\, n$.

2. *TolFit*: If knowledge about the function value of the global minimum $f(\mathbf{x}_{\min})$ is available, the algorithm stops when the current best function value drops below $f(\mathbf{x}_{\min}) + \mathit{TolFit}$.

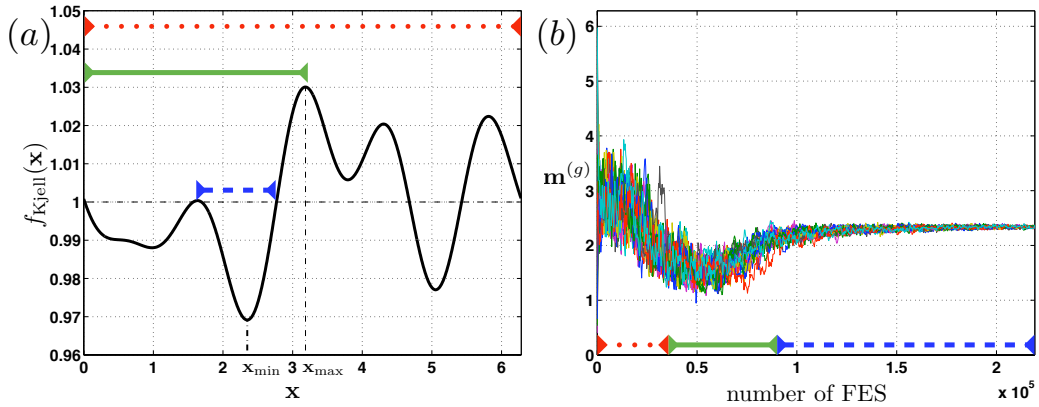



Figure 4.12: (a) The multi-modal function $f_{\mathrm{Kjell}}$ in 1D. The global minimum $\mathbf{x}_{\mathrm{min}}$ is contained in a locally convex region (dashed bar) that belongs to the larger (i.e., maximum entropy) sub-region of the space (solid gray bar). The global maximum $\mathbf{x}_{\mathrm{max}}$ separates this region from the right part of the space. (b) Typical evolution of GaA's mean $\mathbf{m}^{(g)}$ on $f_{\mathrm{Kjell}}$ in $n = 25$. After a global search phase (dotted bar), GaA first adapts a high-entropy distribtion to the broad region (solid gray bar) before it converges to the locally convex global-minimum region (dashed bar).

The default setting is $\mathit{TolFit} = 10^{-9}$.

3. *TolFun*: If no knowledge about $f(\mathbf{x}_{\mathrm{min}})$ is available, GaA is considered converged when $\| \max f(\mathbf{x}^{(i)}) - \min f(\mathbf{x}^{(i)}) \| < \mathit{TolFun}\ \forall i \in [g-h; g]$. By default, we set $\mathit{TolFun} = 10^{-9}$ and the history length $h = 100$.

4. *TolX*: GaA is considered converged when $\|\mathbf{x}^{(g-h)} - \mathbf{x}^{(g)})\| < \mathit{TolX}$. By default, we set $\mathit{TolX} = 10^{-12}$ and the history length $h = 100$.

5. *TolR*: GaA is stopped when the step size $r^{(g)} < \mathit{TolR}$. The default setting is $\mathit{TolR} = 10^{-9}$.

6. *TolCon*: GaA is stopped when the difference between the current threshold $c_{\mathrm{T}}^{(g)}$ and the current best fitness value $f(\mathbf{x}_{\mathrm{best}}^{(g)})$ has converged, i.e., $\|f(\mathbf{x}_{\mathrm{best}}^{(g)}) - c_{\mathrm{T}}^{(g)})\| < \mathit{TolCon}$. The default setting is $\mathit{TolCon} = 10^{-9}$.

These stopping criteria are designed to reduce the number of non-improving function evaluations. They can directly be used to develop an effective restart strategy for GaA as outlined next.

**Restart strategy.** Depending on the landscape topology of the optimization problem, canonical GaA with standard parameter settings may suffer from premature convergence to a sub-optimal solution. This can be relaxed by introducing a restart mechanism that modifies the strategy parameters whenever any of the convergence criteria 3) to 6) above are met. In CMA-ES, restarts with iteratively increasing population size proved powerful both on synthetic and real-world problems. Since GaA always samples a *single* candidate solution per iteration, the

population size cannot be varied. Instead, we adapt the parameter $N_\mathrm{T}$ that controls the lowering of the fitness threshold $c_\mathrm{T}$. The parameters $N_\mathrm{m}$, $N_\mathrm{C}$, $\beta$, and $P$ are kept constant for all restarts. The initial value of $N_\mathrm{T}$ is $N_\mathrm{T}^{(0)} = N_\mathrm{m} = en$. At each restart $i$ we *increase* $N_\mathrm{T}^{(i)}$ as

$$N_\mathrm{T}^{(i)} = r_\mathrm{T} N_\mathrm{T}^{(i-1)} \tag{4.30}$$

with $r_\mathrm{T} = 2$. The new starting point at each restart can either be chosen at random or at the converged position (as in BLR-CMA-ES). The latter strategy is expected to be beneficial on funneled landscapes, such as Rastrigin's or Ackley's function.

The modification of $N_\mathrm{T}^{(i)}$ has a similar effect on GaA as increasing the population size has on CMA-ES. Initially, accepted samples are able to pull down the fitness threshold quickly. On unimodal functions, fast convergence is hence achieved. With increasing $N_\mathrm{T}^{(i)}$, $c_\mathrm{T}$ decreases slower and GaA has more time to explore the space and adapt a maximum-entropy distribution to the underlying landscape structure.

### 4.4.5 Numerical results of Restart GaA on the CEC2005 benchmark suite

We evaluate the performance of Restart GaA again on all 25 CEC 2005 test functions in $n = 10$, 30, and 50 dimensions. These results can directly be compared with the previous data for IPOP-CMA-ES (with correct boundary settings) and PS-CMA-ES. Table 4.10 summarizes the results for Restart GaA on those functions that can be solved within the allowed FES budget.

In $n = 10$ dimensions (Table Table 4.10, upper panel), Restart GaA is able to solve$f_1$to $f_{12}$, except for the needle-in-a-haystack problem f8. IPOP-CMA-ES is able to solve the identical set of functions (see Table 4.2). Functions $f_1$ to $f_7$ are solved with $p_s = 1$. The pair $f_9/f_{10}$ (shifted/rotated Rastrigin) is solved with a lower success probability. Functions $f_{11}$ (shifted Weierstrass) and $f_{12}$ (Schwefel's problem), two multi-funnel functions, are solved with $p_s \geq 0.64$.

In $n = 30$, Restart GaA solves more problems than IPOP-CMA-ES and PS-CMA-ES. It solves $f_1$ to $f_7$, except $f_5$, with $p_s \geq 0.92$, as well as $f_{11}$ with high and $f_{12}$ with low probability. The Rastrigin pair $f_9/f_{10}$ can not be solved any more in 30 dimensions. Similar observations are made for $n = 50$. There, Restart GaA solves as many problems as IPOP-CMA-ES. Problems $f_1$ to $f_4$, $f_7$, and $f_{11}$ can be solved, but neither the Rastrigin pair nor $f_5/f_{12}$ are solved. Closer inspection of the results for $f_6$ (shifted Rosenbrock) reveals that Restart GaA gets close to the minimum, but does not reach the specified accuracy in time.

The invariance of GaA to linear transformations of the search space is verified on the triplet $f_1$–$f_3$, the shifted sphere, Schwefel's problem, and the high-conditional ellipsoid. In $n = 10$ and 30, the mean number of function evaluations used is almost identical for $f_1$ and $f_2$. For $f_3$, GaA needs more samples for properly adapting its covariance matrix. In terms of success performance, these results are outperformed by IPOP-CMA-ES (Auger and Hansen, 2005b). The performance of IPOP-CMA-ES also scales better with the dimensionality $n$, especially

| n=10 | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Func.** | **min** | **median** | **max** | **mean** | **std** | **p$_s$** | **Succ. Perf.** |
| f1 | 7.65e+03 | 8.07e+03 | 8.33e+03 | 8.07e+03 | 1.88e+02 | 1 | 8.07e+03 |
| f2 | 7.80e+03 | 8.31e+03 | 8.56e+03 | 8.25e+03 | 2.05e+02 | 1 | 8.25e+03 |
| f3 | 1.09e+04 | 1.18e+04 | 1.56e+04 | 1.21e+04 | 1.17e+03 | 1 | 1.21e+04 |
| f4 | 7.77e+03 | 8.28e+03 | 1.89e+04 | 8.64e+03 | 2.14e+03 | 1 | 8.64e+03 |
| f5 | 7.28e+03 | 8.20e+03 | - | 1.63e+04 | 1.83e+04 | 0.96 | 1.70e+04 |
| f6 | 1.82e+04 | 2.04e+04 | 2.53e+04 | 2.08e+04 | 1.86e+03 | 1 | 2.08e+04 |
| f7 | 5.11e+03 | 5.46e+03 | 5.90e+03 | 5.45e+03 | 1.91e+02 | 1 | 5.45e+03 |
| f8 | - | - | - | - | - | - | - |
| f9 | 3.74e+04 | - | - | 5.72e+04 | 2.81e+04 | 0.08 | 7.15e+05 |
| f10 | 3.92e+03 | - | - | 4.01e+04 | 3.70e+04 | 0.12 | 3.34e+05 |
| f11 | 1.37e+04 | 4.08e+04 | - | 4.36e+04 | 2.22e+04 | 0.80 | 5.45e+04 |
| f12 | 5.64e+03 | 3.10e+04 | - | 2.61e+04 | 1.77e+04 | 0.64 | 4.08e+04 |
| n=30 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 4.34e+04 | 4.43e+04 | 4.51e+04 | 4.42e+04 | 4.11e+02 | 1 | 4.42e+04 |
| f2 | 4.56e+04 | 4.67e+04 | 4.76e+04 | 4.67e+04 | 4.81e+02 | 1 | 4.67e+04 |
| f3 | 7.19e+04 | 7.93e+04 | 8.97e+04 | 7.91e+04 | 4.44e+03 | 1 | 7.91e+04 |
| f4 | 9.63e+04 | 1.01e+05 | 2.00e+05 | 1.05e+05 | 1.99e+04 | 1 | 1.05e+05 |
| f5 | - | - | - | - | - | - | - |
| f6 | 1.42e+05 | 2.51e+05 | - | 2.47e+05 | 3.63e+04 | 0.92 | 2.68e+05 |
| f7 | 2.98e+04 | 3.05e+04 | 3.17e+04 | 3.06e+04 | 4.73e+02 | 1 | 3.06e+04 |
| f8 | - | - | - | - | - | - | - |
| f9 | - | - | - | - | - | - | - |
| f10 | - | - | - | - | - | - | - |
| f11 | 8.42e+04 | 2.70e+05 | - | 2.25e+05 | 6.04e+04 | 0.80 | 2.81e+05 |
| f12 | 1.75e+05 | - | - | 1.75e+05 | - | 0.04 | 4.37e+06 |
| n=50 | | | | | | | |
| **Func.** | **min** | **median** | **max** | **mean** | **std** | $p_s$ | **Succ. Perf.** |
| f1 | 9.82e+04 | 9.95e+04 | 1.00e+05 | 9.94e+04 | 5.53e+02 | 1 | 9.94e+04 |
| f2 | 1.06e+05 | 1.09e+05 | 1.11e+05 | 1.09e+05 | 1.48e+03 | 1 | 1.09e+05 |
| f3 | 1.94e+05 | 2.04e+05 | 2.20e+05 | 2.06e+05 | 6.97e+03 | 1 | 2.06e+05 |
| f4 | 2.11e+05 | 2.18e+05 | 4.32e+05 | 2.64e+05 | 8.79e+04 | 1 | 2.64e+05 |
| f5 | - | - | - | - | - | - | - |
| f6 | - | - | - | - | - | - | - |
| f7 | 6.63e+04 | 6.77e+04 | 6.87e+04 | 6.77e+04 | 5.52e+02 | 1 | 6.77e+04 |
| f8 | - | - | - | - | - | - | - |
| f9 | - | - | - | - | - | - | - |
| f10 | - | - | - | - | - | - | - |
| f11 | 3.55e+05 | - | - | 4.49e+05 | 5.87e+04 | 0.36 | 1.25e+06 |
| f12 | - | - | - | - | - | - | - |

Table 4.10: Number of FES (min, median, maximum, mean, and standard deviation) for Restart GaA to reach the required accuracy for $f_1$-$f_{12}$ in $n = 10, 30, 50$ within MAX_FES $= 10^4 n$. Columns 7 to 8 give the empirical success rate $p_s$ and the success performance.

for the sphere. Unlike IPOP-CMA-ES, Restart GaA can not solve problem $f_5$, where the global minimum is located at the bounds, for $n \geq 30$. This indicates that we have to revisit the boundary handling mechanism.

The failure of Restart GaA on the Rastrigin pair $f_9/f_{10}$ in $n = 30$ and 50 indicates that population-based methods outperform single-sample strategies on landscapes with many local minima near the global one. Restart GaA, however, shows outstanding robustness against noise as can be derived from its performance on $f_4$. While IPOP-CMA-ES can solve this function in $n = 30$ with $p_s = 0.28$, but fails in $n = 50$, Restart GaA solves it with $p_s = 1$ in all tested dimensions. The noise does not hamper the maximum-entropy adaptation in GaA. It is also noteworthy that the multi-funnel function $f_{11}$ can be efficiently solved by Restart GaA in all tested dimensions. IPOP-CMA-ES solves $f_{11}$ only in $n = 10$. This indicates that $f_{11}$ may have a global landscape structure similar to Kjellström's function, where hierarchical maximum-entropy adaptation is beneficial.

In order to test the efficacy of the proposed new restart procedure, we repeat all tests with $r_T = 1$, i.e., without adapting $N_T$ upon restart (data not shown). For $f_1$ to $f_3$, restart was never needed. In all other cases, doubling $N_T$ upon restart leads to superior performance in *all* problem instances.

### 4.4.6 Conclusions and future work

We have revisited and improved the Gaussian Adaptation algorithm, a scheme for design centering and black-box optimization. Although the basic scheme has already been introduced in the late 1960's and further developed in the 1980's by Kjellström and co-workers, the algorithm has been largely ignored by the optimization community. From a historical perspective, Gaussian Adaptation was the first iterative scheme with covariance matrix adaptation. To the best of our knowledge, it was also the first heuristic that uses threshold acceptance for minimization problems. We contributed several ideas to the basic GaA scheme: First, we showed that the foundation of the algorithm can be derived from Jaynes' Maximum Entropy Principle, hence providing an alternative view on step size adaptation in variable-metric algorithms. Second, we provided suitable standard parameter settings, initialization and boundary handling schemes, as well as an efficient restart strategy. These enhancement render GaA a ready-to-use, parameter-free black-box optimizer. Some basic optimization landscapes have been used to illustrate the search behavior of GaA. Restart GaA has been rigorously tested on the CEC 2005 benchmark test suite. Its remarkable performance ranks the algorithm among the top black-box optimizers ever tested on this benchmark. We expect the optimizer to be especially useful for two kinds of landscapes: those that have a globally unimodal topology corrupted by noise, and multi-modal landscapes that have a global hierarchical structure suitable for maximum-entropy adaptation.

The algorithm with all presented extensions has been implemented as a MATLAB toolbox and is provided to the scientific community as open source package. The technical details can be found in the Appendix A1.

We also emphasize that the Gaussian Adaptation sampling scheme can be used for indirect sampling and high-dimensional volume estimation of convex bodies. While the former application is the topic of the next chapter, the latter is a fundamental problem in computational geometry. Consider a convex body only given via an oracle that can decide for a given sample whether it is inside the body or not. We argue that estimating the volume of a convex body given by a membership oracle is closely related to the design-centering problem (see Fig. 4.7 for illustration). The best currently available convex volume estimators are randomized schemes, such as the Ball-walk and Hit-and-run samplers. We refer to (Vempala, 2005) for a review on this topic. Preliminary numerical experiments suggest that GaA can estimate the volumes of high-dimensional ellipsoidal regions. This is not surprising since knowledge of mean and covariance matrix of a multivariate normal distribution, together with an estimate of the overall hitting probability of the region, suffice to estimate the underlying ellipsoid. For polyhedral bodies, the ellipsoid estimates from GaA can be used as an approximation. Combining our current insights with theoretical investigations will be a topic of future research.

## 4.5 Comparative summary of the benchmark results

We conclude this chapter on black-box optimization by showing the empirical success performance of all tested strategies on the CEC 2005 benchmark test suite in Table 4.11. We present the results for (i) IPOP-CMA-ES as reported in the original publication (Auger and Hansen, 2005b), (ii) IPOP-CMA-ES with and without LD sampling, (iii) PS-CMA-ES with and without LD sampling, and (iv) Restart GaA. Based on the recorded success performances we conclude that IPOP-CMA-ES with LD sampling is superior to all other strategies for the unimodal and moderately multi-modal functions $f_1$–$f_4$ and $f_6$–$f_7$ across all dimensions. For the highly multi-modal functions $f_9/f_{10}$ with globally convex structure IPOP-CMA-ES with LD sampling is superior in $n = 10$ dimensions whereas PS-CMA-ES with LD sampling outperforms all other strategies in $n = 30, 50$ dimensions. For the multi-modal problems $f_{11}$ and $f_{12}$ with weak global structure, Restart GaA shows competitive or superior performance. Standard IPOP-CMA-ES with pseudo-random sampling is competitive or superior across all dimensions only on the unimodal function $f_5$ where the optimum is located at the boundary. Finally, PS-CMA-ES with LD sampling has the best performance on the composite multi-funnel problem $f_{15}$ in $n = 10$.

The presented empirical results demonstrate that the landscape structure has considerable influence on the performance of the different algorithms. This also implies that, if a practitioner has *a priori* knowledge about the landscape structure of the problem he wants to solve, the present results can be used as a guideline for choosing the best suited algorithm. Such landscape knowledge might be gained by calculating the landscape fingerprints presented in the previous chapter prior to optimization. Alternatively, it is also conceivable to design a multi-method algorithm that comprises all presented algorithms and automatically switches between the methods during an optimization run. Switching can be controlled by recording the online progress of an individual algorithm, as successfully demonstrated by Vrugt and coworkers (Vrugt et al., 2009). This paradigm might offer a generic way to combine the strengths of the presented methods in a single general-purpose, parameter-free black-box algorithm.

| n=10 | | | | | | |
|---|---|---|---|---|---|---|
| | **IPOP(orig.)** | **IPOP** | **IPOP(LD)** | **PS-CMA-ES** | **PS-CMA-ES(LD)** | **Restart GaA** |
| f1 | 1.61e+03 | 1.68e+03 | **1.37e+03** | 2.33e+04 | 1.98e+04 | 8.07e+03 |
| f2 | 2.38e+03 | 2.38e+03 | **1.96e+03** | 3.56e+04 | 2.87e+04 | 8.25e+03 |
| f3 | 6.50e+03 | 6.63e+03 | **5.52e+03** | - | 3.07e+05 | 1.21e+04 |
| f4 | 2.90e+03 | 2.56e+03 | **2.08e+03** | 3.86e+04 | 3.02e+04 | 8.64e+03 |
| f5 | 5.85e+03 | 5.76e+03 | **4.79e+03** | 7.93e+04 | 6.68e+04 | 1.70e+04 |
| f6 | 1.08e+04 | 1.05e+04 | **1.00e+04** | 8.20e+04 | 7.28e+04 | 2.08e+04 |
| f7 | 4.67e+03 | 4.11e+03 | **2.94e+03** | 2.41e+04 | 1.97e+04 | 5.45e+03 |
| f8 | - | - | - | - | - | - |
| f9 | 7.57e+04 | 8.87E+04 | **4.78e+04** | 7.04e+04 | 6.30e+04 | 7.15e+05 |
| f10 | 6.50e+04 | 8.52E+04 | **4.68e+04** | 7.66e+04 | 6.75e+04 | 3.34e+05 |
| f11 | 2.63e+05 | 1.83E+05 | **5.29e+04** | 2.86e+05 | 2.57e+05 | 5.45e+04 |
| f12 | 3.27e+04 | 4.48E+04 | **2.00e+04** | 7.79e+04 | 4.81e+04 | 4.08e+04 |
| f13 | - | - | - | - | - | - |
| f14 | - | - | - | - | - | - |
| f15 | - | - | - | 2.29e+06 | **7.79e+05** | - |
| n=30 | | | | | | |
| | **IPOP(orig.)** | **IPOP** | **IPOP(LD)** | **PS-CMA-ES** | **PS-CMA-ES(LD)** | **Restart GaA** |
| f1 | 4.50e+03 | 4.70e+03 | **3.67e+03** | 6.65e+04 | 5.36e+04 | 4.42e+04 |
| f2 | 1.30e+04 | 1.27e+04 | **1.11e+04** | 2.55e+05 | 2.04e+05 | 4.67e+04 |
| f3 | 4.27e+04 | 4.36e+04 | **3.93e+04** | - | - | 7.91e+04 |
| f4 | 5.90e+04 | 2.21e+05 | **5.30e+04** | - | - | 1.05e+05 |
| f5 | 6.59e+04 | **6.57e+04** | 2.55e+05 | 7.39e+06 | - | - |
| f6 | 6.00e+04 | 6.10e+04 | **5.85e+04** | - | - | 2.68e+05 |
| f7 | 6.11e+03 | 6.64e+03 | **5.56e+03** | 7.33e+04 | 5.91e+04 | 3.06e+04 |
| f8 | - | - | - | - | - | - |
| f9 | 7.90e+05 | - | - | 6.56e+06 | **1.36e+06** | - |
| f10 | 2.42e+06 | - | 5.30e+06 | 6.55e+06 | **3.36e+05** | - |
| f11 | 4.98e+06 | - | - | - | - | **2.81e+05** |
| f12 | 2.25e+05 | - | - | - | 5.70e+06 | **4.37e+06** |
| n=50 | | | | | | |
| | **IPOP(orig.)** | **IPOP** | **IPOP(LD)** | **PS-CMA-ES** | **PS-CMA-ES(LD)** | **Restart GaA** |
| f1 | 6.88e+03 | 7.50e+03 | **5.95e+03** | 1.03e+05 | 8.49e+04 | 9.94e+04 |
| f2 | 3.13e+04 | 3.09e+04 | **2.73e+04** | - | 5.00e+05 | 1.09e+05 |
| f3 | 1.17e+05 | 1.18e+05 | **1.09e+05** | - | - | 2.06e+05 |
| f4 | - | - | **1.94e+05** | - | - | 2.64e+05 |
| f5 | - | **2.50e+06** | - | - | - | - |
| f6 | 1.58e+05 | 1.49e+05 | **1.32e+05** | - | - | - |
| f7 | 8.03e+03 | 1.08e+04 | **6.42e+03** | 1.15e+05 | 9.43e+04 | 6.77e+04 |
| f8 | - | - | - | - | - | - |
| f9 | 1.55e+06 | - | - | - | **8.35e+06** | - |
| f10 | 3.76e+06 | - | - | - | **4.47e+06** | - |
| f11 | - | - | - | - | - | **1.25e+06** |
| f12 | - | - | - | - | - | - |

Table 4.11: Success performance of all tested strategies on the CEC 2005 benchmark test suite. The second column gives the originally reported performance of IPOP-CMA-ES (Auger and Hansen, 2005b). Column 3 and 4 give the performance of IPOP-CMA-ES using the pCMALib implementation with pseudo-random and LD sampling, respectively. Columns 5 and 6 give the success performance of PS-CMA-ES with pseudo-random and LD sampling. Column 7 gives the success performance of Restart GaA. The best performance achieved by any of the strategies (except the original IPOP-CMA-ES) is highlighted in bold for each function and dimension.

# 5

# Black-box Sampling: from Landscapes to Probability Distributions

> "Aw, people can come up with statistics to prove anything, Kent. 40% of all people know that."
> Homer Simpson, in: The Simpsons, Homer the Vigilante, Episode no. 92, 1994

## 5.1 Landscapes and probability distributions

We have so far encountered many variations of the landscape concept from evolutionary biology, molecular physics, and optimization. An important prerequisite for characterizing and exploring these landscapes is our ability to draw random samples from certain probability distributions. The uniform and the Gaussian distribution played a crucial role in our previous chapters. Distributions are, however, a ubiquitous concept in their own right in almost all areas of modern science, most prominently in statistics, where observed data $\mathcal{D}$ may be modeled by treating them as samples from a particular distribution $\pi_\theta$. The distribution $\pi_\theta$ is specified up to some parameters $\theta$ that are estimated from the empirical data. With the rise of statistical mechanics at the end of the 19th century, physicists have realized the deep connection between distributions and energy landscapes. Consider the energy landscape of a continuous molecular system, such as a protein surrounded by solvent molecules, consisting of $N$ atoms. The configuration space is $\mathcal{X} \subseteq \mathbb{R}^{3\mathrm{N}}$. Each state $\mathbf{x} \in \mathcal{X}$ has a potential energy $h(\mathbf{x})$, and a distance metric $d(\mathbf{x}, \mathbf{y})$ defines similarity between any two states $\mathbf{x}, \mathbf{y}$. If one is interested in the system behavior at *thermal equilibrium* and constant volume and temperature $T$, statistical mechanics provides a link between the energy landscape and distribution of

states through the concept of a canonical distribution. The example molecular system can be conveniently described by the *Boltzmann distribution* $p_{\mathrm{T}}(\mathbf{x})$:

$$\pi_\theta(\mathbf{x}) = p_{\mathrm{T}}(\mathbf{x}) = \frac{1}{Z(T)} \exp\left(-h(\mathbf{x})/(k_b T)\right). \tag{5.1}$$

In this context, the normalizing constant $Z(T) = \int_{\mathcal{X}} \exp\left(-h(\mathbf{x})/(k_b T)\right) d\mathbf{x} < \infty$ is called the *partition function* of the system. The quantity $k_b$ is a physical constant, such that the temperature is the only parameter $\theta = T$ of the distribution. The canonical distribution thus maps the energy landscape into a probability distribution. Vice versa, if we have knowledge about a certain (not necessarily physical) probability distribution $\pi_\theta(\mathbf{x})$, we can transform it into a generalized energy $h(\mathbf{x})$ through:

$$h(\mathbf{x}) = -\log \pi_\theta(\mathbf{x}) - \log Z\,, \tag{5.2}$$

where we arbitrarly set $k_b = T = 1$, and $Z$ is any convenient positive constant. Zhou and Wong's recent article "Reconstructing the energy landscape of a distribution from Monte Carlo samples" exploits this relationship and demonstrates the benefits of the landscape perspective for statistical inference (Zhou and Wong, 2008). Liu emphasizes that presumably all probability distributions can be recast into the form: $\pi(\mathbf{x}) = 1/Z \exp\left(-h(\mathbf{x})\right)$ (Liu, 2002).

The strict monotonicity of the canonical transformation implies that the global topology of the energy landscape is conserved in the resulting distribution. For instance, a multi-modal energy landscape induces a multi-modal distribution, and vice versa. All our previous considerations about landscape topologies and landscape features can, hence, be carried over to the realm of probability distributions. We will come back to this important point in the next section.

In the previous chapter, we have been largely concerned with identifying regions of low energy or objective function values in a landscape. In the context of probability distributions the main objective is different: Given a probability distribution, we want to *generate unbiased samples* from this distribution in order to infer knowledge about the underlying system that it describes. For example, we may want to estimate moments of the distribution such as expectation values or derived averages. In a physical context, these averages are often termed *macro states* as they are related to measurable observables in a real experiment. *Direct sampling* methods to generate samples are available for a number of distributions, including the Gaussian and the uniform distribution. However, for a large class of distributions such schemes do not exist, and one has to rely on *indirect sampling* or *black-box sampling* procedures. Consider again the above system described by the Boltzmann distribution. In principle, this distribution is completely specified for all micro states $\mathbf{x}$. A crucial caveat, however, is the normalizing constant $Z(T)$. Computating this high-dimensional integral is usually intractable. Hence, only a function $f_{\mathrm{T}}(\mathbf{x}) \propto p_{\mathrm{T}}(\mathbf{x})$ can be evaluated for all $\mathbf{x} \in \mathcal{X}$. This situation is by no means a singular phenomenon. It also occurs in many Bayesian approaches to statistical inference. There, the posterior distribution for the model parameters is the focus of interest which, in most cases, can again only be specified up to a normalizing constant. This situation leaves us with two important questions for this chapter: (i) How can we generate unbiased samples from such distributions in a black-box fashion, i.e., only through knowledge about the

absolute frequencies of sampled configurations, and (ii) how are these approaches related to the presented Black-box optimization schemes? While the answer to the first question can be traced back to one of the earliest studies in computational science, the answer to the second question is a key result of this thesis.

## 5.2 Black-box sampling using Markov chains

A large class of black-box samplers is based on the generation of so-called *Markov chains*. We thus give a short introduction to Markov chains and trace the development of Markov-Chain Monte Carlo (MCMC) methods. Equipped with these preliminaries we introduce adaptive MCMC methods that are conceptually very close to GaA. The synthesis of both paradigms then leads to a novel adaptive MCMC method, the Metropolis GaA algorithm.

### 5.2.1 Markov-Chain Monte Carlo methods

Consider a sequence of random variables $\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(t)}, \ldots, \mathbf{x}^{(N)}$ defined on a finite state space $\mathcal{X}$. This sequence is called a Markov chain, named after the Russian mathematician Andrey Markov, if it satisfies the *Markov property*:

$$P(\mathbf{x}^{(t+1)} = \mathbf{y} \,|\, \mathbf{x}^{(t)} = \mathbf{x}, \ldots, \mathbf{x}^{(0)} = \mathbf{z}) = P(\mathbf{x}^{(t+1)} = \mathbf{y}) | \mathbf{x}^{(t)} = \mathbf{x}) \,, \tag{5.3}$$

with $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{X}$. This means that the probabaility of next state $\mathbf{x}^{(t+1)}$ *only* depends on the current state $\mathbf{x}^{(t)}$. $P(\cdot|\cdot)$ hence denotes the *transition probability* from a given state to the next one. If $P(\cdot|\cdot)$ is time-homogeneous, i.e., it does not change with $t$, it is often expressed as a transition function $A(\mathbf{x}, \mathbf{y})$ that has the simple property

$$\sum_{\mathbf{y}} A(\mathbf{x}, \mathbf{y}) = 1 \quad \forall \, \mathbf{y} \in \mathcal{X} \,. \tag{5.4}$$

When the state space $\mathcal{X}$ is continuous, the transition probability function is replaced by a *transition density function* and summation is replaced by integration.

Recall our objective of drawing unbiased samples from a target distribution $\pi(\mathbf{x})$ defined on $\mathcal{X}$. The fundamental idea of MCMC methods is to simulate a Markov chain in $\mathcal{X}$ such that the *stationary* distribution of this chain is the target distribution $\pi(\mathbf{x})$. This objective can be recast into the problem of finding a transition function $A(\mathbf{x}, \mathbf{y})$ to which $\pi(\mathbf{x})$ is invariant. The mathematical formulation of this statement is:

$$\int_{\mathcal{X}} \pi(\mathbf{x}) A(\mathbf{x}, \mathbf{y}) d\mathbf{x} = \pi(\mathbf{y}) \,. \tag{5.5}$$

A simpler, yet more restrictive condition than Eq. (5.5) is the so-called *detailed balance* condition:

$$\pi(\mathbf{x}) A(\mathbf{x}, \mathbf{y}) = \pi(\mathbf{y}) A(\mathbf{y}, \mathbf{x}) \,. \tag{5.6}$$

This can be verified by substituting Eq. (5.6) into Eq. (5.5):

$$\int_{\mathcal{X}} \pi(\mathbf{x}) A(\mathbf{x}, \mathbf{y}) d\mathbf{x} = \int_{\mathcal{X}} \pi(\mathbf{y}) A(\mathbf{y}, \mathbf{x}) d\mathbf{x} = \pi(\mathbf{y}) \int_{\mathcal{X}} A(\mathbf{y}, \mathbf{x}) d\mathbf{x} = \pi(\mathbf{y}) \,, \tag{5.7}$$

where the last step follows from Eq. (5.4). Hence, detailed balance is a sufficient, but not a necessary condition for invariance. Markov chains that satisfy detailed balance are called *reversible*.

In their seminal paper "Equation of State Calculation by Fast Computing Machines" (Metropolis et al., 1953) Nicholas Metropolis and co-workers introduced an algorithm, later named the Metropolis algorithm, that started the entire field of MCMC methods. Being in the comfortable position of having access to one of the first computers, the MANIAC system at the Los Alamos National Laboratory, the authors developed a method for calculating thermodynamic quantities through simulations of ensembles of interacting molecules. In the original article a collection of $N$ particles in the unit plane with periodic boundary conditions was considered. The state space hence was $\mathcal{X} = [0, 1]^{2N}$. Denoting the position of the $i^{\text{th}}$ particle by $(x_i, y_i)$, each configuration $\mathbf{x} = [x_1, x_2, \ldots, x_N, y_1, y_2, \ldots, y_N]$ of particles, a potential energy $h(\mathbf{x})$ could be calculated. The algorithm starts from an initial random particle configuration $\mathbf{x}^{(0)}$ and iteratively updates the configuration. The algorithm terminates after $K$ rounds. In a round each particle is sequentially selected. The position of particle $i$ is moved according to:

$$x_i' = x_i + c\eta\,, \qquad y_i' = y_i + c\eta \tag{5.8}$$

where $c$ is a scalar constant for the maximum allowed displacement and $\eta \sim \mathcal{U}[-1, 1]$. The energy $h(\mathbf{y})$ of the new configuration $\mathbf{y} = [x_1, x_2, \ldots, x_i', \ldots, x_N, y_1, y_2, \ldots, y_i', \ldots, y_N]$ is calculated, and the following expression is computed:

$$\alpha_{\mathrm{M}}(\mathbf{y}, \mathbf{x}^{(t)}) = \min\left(1, \frac{\exp\left(-h(\mathbf{y})/(k_b T)\right)}{\exp\left(-h(x^{(t)})/(k_b T)\right)}\right). \tag{5.9}$$

The new configuration is $\mathbf{x}^{(t+1)} = \mathbf{y}$ if $h(\mathbf{y}) \leq h(\mathbf{x}^{(t)})$ or $\psi \leq \alpha(\mathbf{y}, \mathbf{x}^{(t)})$ for $\psi \sim \mathcal{U}[0, 1]$. Otherwise, $\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)}$. One round is completed when all $N$ particles have been subject to an attempted move, hence $t = 0, ..., KN$. We call this method the Single Component (SC) Metropolis algorithm because each component (particle) is moved independently. Metropolis and co-workers showed that this scheme satisfies detailed balance and the generated samples hence yield unbiased estimates of thermodynamic quantities of the system. The Metropolis algorithm is neither restricted to single-component nor to uniformly distributed moves. In fact, the only requirement on the move set or *proposal distribution* $q(\cdot|\cdot)$ is symmetry, i.e. $q(\mathbf{x}|\mathbf{y}) = q(\mathbf{y}|\mathbf{x})$. Green and Han (Green and Han, 1992) were among the first to use the isotropic Gaussian distribution as $q(\cdot|\cdot)$ for continuous target distributions. The Metropolis algorithm with a Gaussian proposal distribution is often termed the Standard Random Walk Metropolis sampling algorithm (Liu, 2002) or Normal Symmetric Random Walk Metropolis (N-SRWM) algorithm (Andrieu and Thoms, 2008). It is straightforward to prove that any Metropolis algorithm with symmetric proposals satisfies detailed balance. Given the target distribution in the general form $\pi(\mathbf{x}) = 1/Z \exp\left(-h(\mathbf{x})\right)$, the transition density function for the general Metropolis algorithm is:

$$A(\mathbf{x}, \mathbf{y}) = q(\mathbf{x}|\mathbf{y}) \min\left(1, \frac{\pi(\mathbf{y})}{\pi(\mathbf{x})}\right), \tag{5.10}$$

where the second term is the general *Metropolis acceptance criterion*. For this transition rule the detailed balance condition is satisfied for any symmetric $q(\cdot|\cdot)$:

$$\begin{aligned}\pi(\mathbf{x})A(\mathbf{x},\mathbf{y}) &= \pi(\mathbf{x})q(\mathbf{x}|\mathbf{y})\min\left(1,\frac{\pi(\mathbf{y})}{\pi(\mathbf{x})}\right) = \min\left(\pi(\mathbf{x})q(\mathbf{x}|\mathbf{y}),\pi(\mathbf{y})q(\mathbf{x}|\mathbf{y})\right) \\ &= \min\left(\pi(\mathbf{x})q(\mathbf{y}|\mathbf{x}),\pi(\mathbf{y})q(\mathbf{y}|\mathbf{x})\right) = \pi(\mathbf{y})q(\mathbf{y}|\mathbf{x})\min\left(\frac{\pi(\mathbf{x})}{\pi(\mathbf{y})},1\right) = \pi(\mathbf{y})A(\mathbf{y},\mathbf{x})\,,\end{aligned} \tag{5.11}$$

which proves the statement. The Metropolis algorithm also gained considerable popularity in statistics through W. Keith Hasting's landmark paper "Monte Carlo sampling methods using Markov chains and their applications" (Hastings, 1970). Hasting generalized the Metropolis algorithm to asymmetric proposal densities $q(\mathbf{x}|\mathbf{y}) \neq q(\mathbf{y}|\mathbf{x})$, leading to the Metropolis-Hastings acceptance criterion:

$$\alpha_{\mathrm{MH}}(\mathbf{x},\mathbf{y}) = \min\left(1,\,\frac{\pi(\mathbf{y})q(\mathbf{y}|\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x}|\mathbf{y})}\right)\,. \tag{5.12}$$

The proof that this new transition density also satisfies detailed balance is analogous to Eq. (5.11). Due to this generalization the algorithm is nowadays often called the *Metropolis-Hastings* MH algorithm independent of whether a symmetric or asymmetric proposal density is used.

Over the past 40 years a tremendous number of variations of the MH algorithm has been developed. The Metropolized Independence sampler is a variation where the proposal does not depend on the current state of the Markov chain (Hastings, 1970). The ubiquitous Gibbs sampler (Geman and Geman, 1984), another special version of the MH algorithm, is applicable when the target distribution is not known explicitly, but only the conditional distribution of each variable is known. For a superb overview of MH algorithms we refer to the book of Liu (Liu, 2002). In the following we focus on the popular N-SRWM algorithm where the next state of the Markov chain, given the current state $\mathbf{x}^{(g)}$, is sampled from $\mathcal{N}\big(\mathbf{x}^{(g)}, c_n^2\mathbf{I}_n\big)$. $\mathbf{I}_n$ is the $n$-dimensional identity matrix and $c_n$ is the scalar, $n$-dependent standard deviation. The choice of $c_n$ is crucial for the efficiency of the sampling process. Gelman and co-workers empirically showed that for $n$-dimensional Gaussian target distributions, the optimal choice is $c_n = 2.4/\sqrt{n}$ (Gelman et al., 1996). This corresponds to an optimal overall acceptance rate of $P^* \approx 0.234$, i.e., among all proposed states approximately 23% are accepted by the sampler. Many empirical studies confirmed that this choice is also usable for arbitrary target distributions.

It is fascinating that this value is very close to Rechenberg's 1/5-rule for the (1+1)-ES. The performance of the N-SRWM algorithm can, however, deteriorate when the target distribution has a highly irregular shape or has a multi-funnel character. Loss of performance often means that the Markov chain gets stuck in some state $\mathbf{x}^{(g)}$ where new proposal states are almost always rejected. Another frequently encountered behavior is that the Markov chain explores only a small portion of the total state space (for instance, one out of many funnels), causing bias in the estimates of macroscopic quantities. In practice, researchers thus often perform preliminary runs using different shapes and scales of the covariance and assess the acceptance

rate and the"mixing" of the chain. Production runs are then conducted with the fine-tuned Gaussian proposal. If prior knowledge about the covariance matrix $\mathbf{C}$ or at least a rough estimate of the target distribution is available, the proposal is often changed to $\mathcal{N}\big(\mathbf{x}^{(g)}, c_n^2\mathbf{C}\big)$, which can significantly accelerate the mixing of the chain. Nonetheless, it is conceivable that the use of a static proposal distribution throughout an entire MCMC run is not always the best choice. It is a surprising fact that the idea of self-adaptation of the proposal distribution, a long standing mechanism in Evolution Strategies, has not been recognized in the statistics community until the late 1990's. The Adaptive Proposal (AP) algorithm by Haario and co-workers (Haario et al., 1999) is probably the first algorithm of this new class of *Adaptive Markov-Chain Monte Carlo methods*.

### 5.2.2 Adaptive Markov-Chain Monte Carlo

Adaptive Markov-Chain Monte Carlo methods have been introduced in order to avoid the difficulties of fine tuning the proposal distributions in MH algorithms. This is conceptually related to the self-adaptation ideas in ES and GaA. An adaptive MCMC method is allowed to *learn* a better proposal based on the information provided by previous sample points. We first outline the AP algorithm and its generalization, the Adaptive Metropolis (AM) algorithm (Haario et al., 2001), before presenting the generic framework of adaptive MCMC.

The MH algorithm constructs a chain that fulfills the Markov property given in Eq. (5.3). States at iteration $(g+1)$ are sampled from $q(\cdot|\mathbf{x}^{(g)})$ and the history of accepted points is discarded. The Adaptive Proposal (AP) algorithm (Haario et al., 1999) uses a Gaussian proposal distribution that depends on $H$ previously accepted points, hence $q(\cdot|\mathbf{x}^{(g)}, \ldots, \mathbf{x}^{(g-H+1)})$. This history $H$ is used to continuously adapt the covariance matrix $\mathbf{C}^{(g)}$ of the proposal density, i.e. $q(\cdot|\mathbf{x}^{(g)}, \ldots, \mathbf{x}^{(g-H+1)}) \sim \mathcal{N}\big(\mathbf{x}^{(g)}, c_n^2\mathbf{C}^{(g)}\big)$. $\mathbf{C}^{(g)}$ is calculated by collecting the $H$ states $\mathbf{x}^{(g)}, \ldots, \mathbf{x}^{(g-H+1)}$ in the $H \times n$ history matrix $\mathbf{A}$, where each row represents one sampled state. Then

$$\mathbf{C}^{(g)} = \frac{1}{H-1}\hat{\mathbf{A}}^T\hat{\mathbf{A}} \tag{5.13}$$

with the centered history matrix $\hat{\mathbf{A}} = \mathbf{A} - E[\mathbf{A}]$. Samples from $\mathcal{N}\big(\mathbf{x}^{(g)}, c_n^2\mathbf{C}^{(g)}\big)$ are drawn using:

$$\mathcal{N}\Big(\mathbf{x}^{(g)}, c_n^2\mathbf{C}^{(g)}\Big) \sim \mathbf{x}^{(g)} + \frac{c_n}{\sqrt{H-1}}\hat{\mathbf{A}}^T\mathcal{N}(\mathbf{0}, \mathbf{I}_{\mathrm{H}}) \ , \tag{5.14}$$

where $\mathbf{I}_{\mathrm{H}}$ is the $H$-dimensional identity matrix. This clever relationship avoids the computation of the Cholesky decomposition of $\mathbf{C}^{(g)}$ at the expense of generation $H$ instead of $n$ standard Gaussian variates. We come back to this idea in the last part of this chapter. Besides the scalar parameter $H$, Haario and co-workers also decided to update the covariance not at every step $g$ but with a certain update frequency $U$. That means that only after $U$ steps the accumulated information about the previously accepted states is utilized for covariance update. They suggest an approximately linear scaling of $H$ and $U$ with dimension $n$, for instance $H = U = 200$ for $n = 2$ and $H = U = 700$ for $n = 8$ (see Tab. 4 in (Haario et al., 1999) for details). We present numerical experiments with AP in Section 5.3 showing improved mixing of the chain and hence superior performance on several target distributions when compared to MH. The AP algorithm produces a time-inhomogeneous Markov chain,

thus violating reversibility and the detailed balance condition. A proof of ergodicity is not available for AP. However, slight modifications of the AP scheme, together with restrictions on the class of target distributions, enabled theoretical results for the Adaptive Metropolis (AM) algorithm (Haario et al., 2001). The AM algorithm is equivalent to an AP algorithm with complete sample history and a regularization term for the covariance matrix. The AM scheme divides the sampling process into two phases, an initial static sampling and an adaptive phase:

$$\mathbf{C}^{(g)} = \begin{cases} \mathbf{C}^{(0)}, & \text{if } g \leq g_0 \\ c_n^2 cov\left(\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(g-1)}\right) + c_n^2 \epsilon \, \mathbf{I}_n, & \text{if } g > g_0 \,. \end{cases} \tag{5.15}$$

$\mathbf{C}^{(0)}$ is a user-defined initial covariance matrix and $cov\left(\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(g-1)}\right)$ is the empirical covariance estimated from the full history of accepted samples:

$$cov\left(\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(g-1)}\right) = \frac{1}{g-1}\left(\sum_{i=0}^{g} \mathbf{x}^{(i)}\mathbf{x}^{(i)\,T} - (g+1)\hat{\mathbf{x}}^{(g)}\hat{\mathbf{x}}^{(g)\,T}\right) \tag{5.16}$$

with $\hat{\mathbf{x}}^{(g)} = \frac{1}{g+1}\sum_{i=0}^{g} \mathbf{x}^{(i)}$. The term $c_n^2 \epsilon \, \mathbf{I}_n$ with $\epsilon > 0$ can be seen as a Tikhonov regularizer (as in ridge regression), ensuring regularity of the covariance matrix even in cases where the empirical covariance $cov\left(\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(g-1)}\right)$ becomes singular. Haario and co-workers also present a recursive formula for $g > g_0$ for the covariance update:

$$\begin{aligned} \mathbf{C}^{(g+1)} = {} & \frac{g-1}{g}\mathbf{C}^{(g)} \\ & + \frac{c_n^2}{g}\left(g\,\hat{\mathbf{x}}^{(g-1)}\hat{\mathbf{x}}^{(g-1)\,T} - (g+1)\hat{\mathbf{x}}^{(g)}\hat{\mathbf{x}}^{(g)\,T} + \mathbf{x}^{(g)}\mathbf{x}^{(g)\,T} + \epsilon\,\mathbf{I}_n\right) . \end{aligned} \tag{5.17}$$

This recursion constitutes a rank-one update of the proposal covariance matrix with decreasing weights. For $g \to \infty$ adaptation of the covariance vanishes, a property that is essential in the proof of ergodicity for the AM algorithm. Haario and co-workers could show that, for target distributions $\pi(\mathbf{x})$ that are bounded from above and are defined on a *bounded* support the samples from the AM algorithm with $\epsilon > 0$ yield $\pi(\mathbf{x})$ as the limiting distribution. The proof is very technical, involving ideas from the theory of *mixingales*, a specific class of stochastic processes (McLeish, 1975). Although the performance of the AM and AP algorithm is comparable in pratice, the AM algorithm has been immensely valuable as it constitutes the first MCMC method where ergodicity has been proven, even though the chain is nowhere Markovian in the sense of Eq. (5.3). Throughout the past 10 years, the AM algorithm triggered a tremendous amount of both theoretical and practical studies on adaptive MCMC methods (Andrieu and Moulines, 2006). Andrieu and Robert realized that the AM algorithm is a special instance of the general class of adaptive or *controlled MCMC* methods (Andrieu and Robert, 2001). *Stochastic approximation* algorithms of the Robbins-Monro type (Robbins and Monro, 1951) and simulated tempering algorithms (Marinari and Parisi, 1992; Geyer and Thompson, 1995) also belong to this class. In a recent article Andrieu and Thoms provide a generic framework for adaptive MCMC methods (Andrieu and Thoms, 2008). They radically reduce adaptive MCMC methods to their key ingredients. One such ingredient is the concept of "vanishing adaptation". They claim that this is fundamental for consistent adaptive MCMC. The AM

algorithm satisfies vanishing adaptation by making the covariance less and less dependent on the actual samples (see Eq. (5.17)). Furthermore, Andrieu and Thoms remark that the covariance is not the only quantity that can be subject to control or adaptation. They provide generic instances of adaptive MCMC algorithms based on the N-SRWM algorithm. The Gaussian proposal density $q_\theta(\cdot|\cdot)$ has three adaptable parameters $\theta = \{\mathbf{m}, r, \mathbf{C}\}$: the mean of the proposal, a scale factor, and the covariance. The generalized AM algorithm adapts the mean and the covariance as shown in Alg. 3

**Algorithm 3**: General AM algorithm

**Input**: Initial $\mathbf{x}^{(0)}$, $\mathbf{m}^{(0)}$, $\mathbf{C}^{(0)}$, $r$
**Result**: Unbiased sample $\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(K)}$ from target distribution $\pi(\mathbf{x})$
**for** $g = 0, 1, \ldots, K-1$ **do**

1. Sample $\mathbf{x}^{(g+1)} \sim \mathcal{N}\left(\mathbf{x}^{(g)}, r^2\mathbf{C}^{(g)}\right)$
2. Apply Metropolis criterion $\alpha_{\mathrm{M}}(\mathbf{x}^{(g+1)}, \mathbf{x}^{(g)}) = \min\left(1, \frac{\pi(\mathbf{x}^{(g+1)})}{\pi(\mathbf{x}^{(g)})}\right)$
3. Update

$$\mathbf{m}^{(g+1)} = \mathbf{m}^{(g)} + \gamma^{(g+1)}\left(\mathbf{x}^{(g+1)} - \mathbf{m}^{(g)}\right)$$

$$\mathbf{C}^{(g+1)} = \mathbf{C}^{(g)} + \gamma^{(g+1)}\left(\left(\mathbf{x}^{(g+1)} - \mathbf{m}^{(g)}\right)\left(\mathbf{x}^{(g+1)} - \mathbf{m}^{(g)}\right)^T - \mathbf{C}^{(g)}\right)$$

**end**

Andrieu and Thoms provide conditions that need to be fulfilled for the sequence $\gamma^{(g)}$ to achieve vanishing adaptation. From these conditions one can see that any sequence of the form $\gamma^{(g)} = \gamma_0/g^k$ with $k \in [(1+\epsilon)^{-1}, 1]$, and with $\gamma_0, \epsilon > 0$ is consistent. In the original AM algorithm Haario and co-workers simply used $\gamma^{(g)} = 1/g$. Andrieu and Thoms also provide alternative update formulae for the mean and the covariance (see (Andrieu and Thoms, 2008) for details). Moreover, they suggest an AM variant that also adapts the scale factor $r$. They term this generic algorithm *the AM algorithm with global adaptive scaling*, see Algorithm 4. Adaptation of the global scale factor $r$ provides a way of controlling the acceptance rate of the sampler. $P^*$ is the user-defined, fixed target acceptance rate of the sampler. Gelman's optimal rate of 0.234 for Gaussian targets could be a default choice.

It is amazing that this scheme coincides *exactly* with the generic framework of GaA and CMA-ES-like optimization algorithms. Note that in Algorithm 4 the update of the covariance and the global step size are not decoupled (like in CMA-ES), thus hampering an efficient learning of the global scale in this context. As a remedy Andrieu and Thoms presented variations of this scheme, such as component-wise update of the mean, covariance, and scale factors. All of these have well-known equivalents in the optimization world for many years. We emphasize that this close relationship between the black-box optimizers and adaptive MCMC methods has previously not been recognized. We comment on the possible cross-fertilization of both fields in the last part of this chapter. The next section introduces a GaA variant for black-box

**Algorithm 4**: General AM algorithm with global adaptive scaling

**Input**: Initial $\mathbf{x}^{(0)}$, $\mathbf{m}^{(0)}$, $\mathbf{C}^{(0)}$, $r^{(0)}$ and $P^*$
**Result**: Unbiased sample $\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(K)}$ from target distribution $\pi(\mathbf{x})$
**for** $g = 0, 1, \ldots, K-1$ **do**

1. Sample $\mathbf{x}^{(g+1)} \sim \mathcal{N}\left(\mathbf{x}^{(g)}, r^{(g)\,2}\mathbf{C}^{(g)}\right)$
2. Apply Metropolis criterion $\alpha_{\mathrm{M}}(\mathbf{x}^{(g+1)}, \mathbf{x}^{(g)}) = \min\left(1, \frac{\pi(\mathbf{x}^{(g+1)})}{\pi(\mathbf{x}^{(g)})}\right)$
3. Update

$$\log\left(r^{(g+1)}\right) = \log\left(r^{(g)}\right) + \gamma^{(g+1)}\left(\hat{\alpha}_{\mathrm{M}}(\mathbf{x}^{(g+1)}, \mathbf{x}^{(g)}) - P^*\right)$$

$$\mathbf{m}^{(g+1)} = \mathbf{m}^{(g)} + \gamma^{(g+1)}\left(\mathbf{x}^{(g+1)} - \mathbf{m}^{(g)}\right)$$

$$\mathbf{C}^{(g+1)} = \mathbf{C}^{(g)} + \gamma^{(g+1)}\left(\left(\mathbf{x}^{(g+1)} - \mathbf{m}^{(g)}\right)\left(\mathbf{x}^{(g+1)} - \mathbf{m}^{(g)}\right)^T - \mathbf{C}^{(g)}\right)$$

**end**

sampling, which can be seen as a specific instance of an adaptive MCMC algorithm with global adaptive scaling.

### 5.2.3 Metropolis Gaussian Adaptation: an adaptive MCMC method

Gaussian Adaptation provides a unifying framework for black-box optimization and adaptive MCMC. We call the adaptive MCMC version of GaA *Metropolis Gaussian Adaptation* (M-GaA). When using GaA for optimization, sample points with function values higher than the threshold $c_{\mathrm{T}}$ are strictly rejected and points with lower values accepted. For black-box sampling this hard threshold is replaced with the Metropolis acceptance-rejection scheme $\alpha_{\mathrm{M}}(\mathbf{x}^{(g+1)}, \mathbf{x}^{(g)}) = \min\left(1, \frac{\pi(\mathbf{x}^{(g+1)})}{\pi(\mathbf{x}^{(g)})}\right)$. In cases where the continuous target probability distribution $\pi(\mathbf{x})$ is only known up to a normalization constant, $f(\mathbf{x}) \propto \pi(\mathbf{x})$ is used in the acceptance criterion. The recursive rule for the threshold adaptation in Eq. (4.27) is obsolete in M-GaA. We further set the weight parameter $N_{\mathrm{m}} = 1$, moving GaA's mean directly to the accepted sample $\mathbf{x}^{(g+1)}$. For the remaining parameters the standard settings are used. This yields a sampling algorithm with adaptive Gaussian proposals. Moreover, M-GaA possesses the convenient feature of setting the acceptance probability $P$ *a priori*. If not stated otherwise, the standard settings is $P = 0.234$. This renders M-GaA an adaptive MCMC sampler with global adaptive scaling *and* decoupling of covariance orientation and scale because the updated covariance is constantly normalized. Like the AP algorithm, M-GaA does not yet embed the concept of vanishing adaptation, leading to a scheme for which ergodicity can probably not be proven. In principle, it is straightforward to satisfy vanishing adaptation in M-GaA. However, at this point we are interested in the practical performance of the standard GaA scheme when used as a sampler.

## 5.3 Computational experiments

We consider two different test scenarios for M-GaA: The first one is inspired by the tests used in Haario's publication of the AP algorithm (Haario et al., 1999) and in (Andrieu and Thoms, 2008). The second one presents a benchmark from Neal's article on Slice Sampling (Neal, 2003), showing the limitations of the current M-GaA scheme.

### 5.3.1 Haario's distributions

In order to assess the performance of M-GaA as an adaptive sampler, we first follow the protocol outlined in (Haario et al., 1999). We consider the same three test target distributions:

$\pi_1$: Uncorrelated Gaussian distribution

$\pi_2$: Moderately twisted Gaussian distribution

$\pi_3$: Strongly twisted Gaussian distribution

Distribution $\pi_1$ is a centered $n$-dimensional multivariate normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{C_1})$ with $\mathbf{C_1} = diag(100, 1, \ldots, 1)$. It thus has the shape of an axis-aligned hyper-ellipsoid with an axes aspect ratio of 10. The twisted Gaussians are constructed as follows: Let $g$ be the density of an uncorrelated Gaussian. The density function of a twisted Gaussian with twisting parameter $b > 0$ is then given by

$$g_b = g(\Phi_b(\mathbf{x}))\,, \tag{5.18}$$

where $\Phi_b(\mathbf{x}) = (x_1, x_2 + bx_1^2 - 100b, x_3, \ldots, x_n)$. $\Phi_b$ thus only affects the second coordinate, and the determinant of its Jacobian is unity (Haario et al., 1999). It is easy to compute probability regions of $g_b$ and to verify that the expectation value of $g_b$ is $\mathbf{0}$ for all $b$. Haario et al. used $b = 0.03$ for $\pi_2$ and $b = 0.1$ for $\pi_3$. Figure 5.1 shows the contour lines of the 68.3% and 99% probability regions of $\pi_1$ to $\pi_3$. Haario et al. also suggested the following quality measures for sampling algorithms:

1. mean($\|E\|$): The mean distance of the expectation values from their true value ($\mathbf{0}$), averaged over $N$ repetitions

2. std($\|E\|$): The standard deviation of the distance of the expectation values from their true value, averaged over $N$ repetitions

3. err($\leq 68.3\%$): The mean error (in %) of the percentage of sampled points that hit the probability region inside the 68.3% contour

4. std($\leq 68.3\%$): The standard deviation of err($\leq 68.3\%$)

5. err($> 99\%$): The mean error (in %) of the percentage of sampled points that hit the probability region outside the 99% contour.

6. std($> 99\%$): The standard deviation of err($> 99\%$)

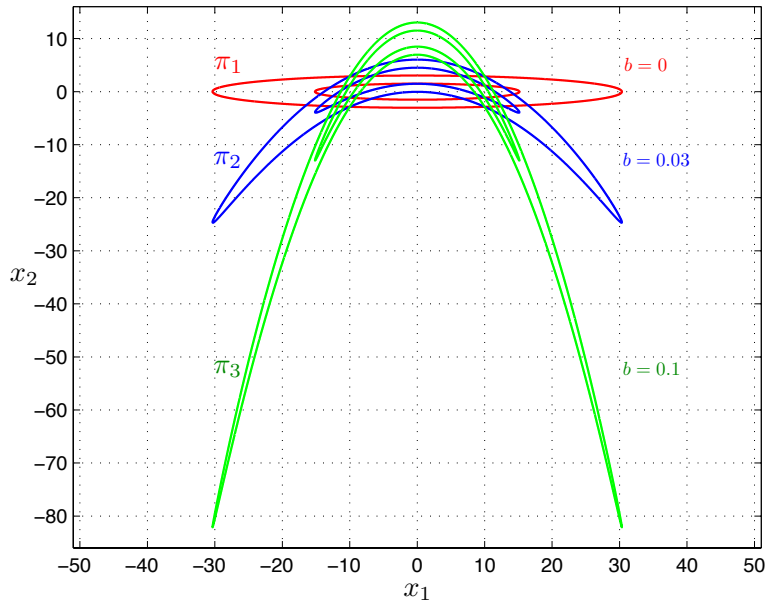



Figure 5.1: 68.3% and 99% probability regions of the three test target distributions $\pi_1$ (red), $\pi_2$ (blue), and $\pi_3$ (green) in 2D. The parameter $b$ controls the isochoric distortion of the Gaussian density (see main text for details).

We test the M-GaA sampling scheme on the target distributions $\pi_1$ to $\pi_3$ in $n = 8$ dimensions (Haario et al., 1999). 100 independent runs are performed for each target. The sample size is limited to 20000 for $\pi_1$, 40000 for $\pi_2$, and 80000 for $\pi_3$. In all test cases, M-GaA's initial sample point is drawn uniformly at random within the hypercube $[-1, 1]^8$, and the initial step size is $r^{(0)} = 1$. Since the chain in M-GaA rapidly mixes, the burn-in length is set to $1,000$. In order to be close the empirical acceptance probability of the AP algorithm (0.14 for $\pi_2$ and 0.09 for $\pi_3$ (Haario et al., 1999)), the hitting probability $P$ is set to 0.1 in all cases. Figure 5.2 shows 2D projections of some M-GaA samples from each target distribution. We compare the performance of M-GaA to three other algorithms:

1. Single-component Metropolis algorithm (SC) with univariate Gaussian proposal. This algorithm explores each coordinate axis separately, one after the other (Metropolis et al., 1953).

2. Metropolis-Hastings algorithm (MH) with isotropic multivariate Gaussian proposal (i.e. the N-SRWM scheme). This algorithm explores all directions simultaneously.

3. Adaptive Proposal algorithm. Note that the AM algorithm achieves similar performance as AP on these test targets (see (Haario et al., 2001) for details).

For both SC and MH, the standard deviation of the Gaussian proposal is fixed to the optimal value of $2.4/\sqrt{n}$. For AP we use the parameter values given in (Haario et al., 1999). The burn-in length is set to 50% of the sample size for all three algorithms (Haario et al., 1999).

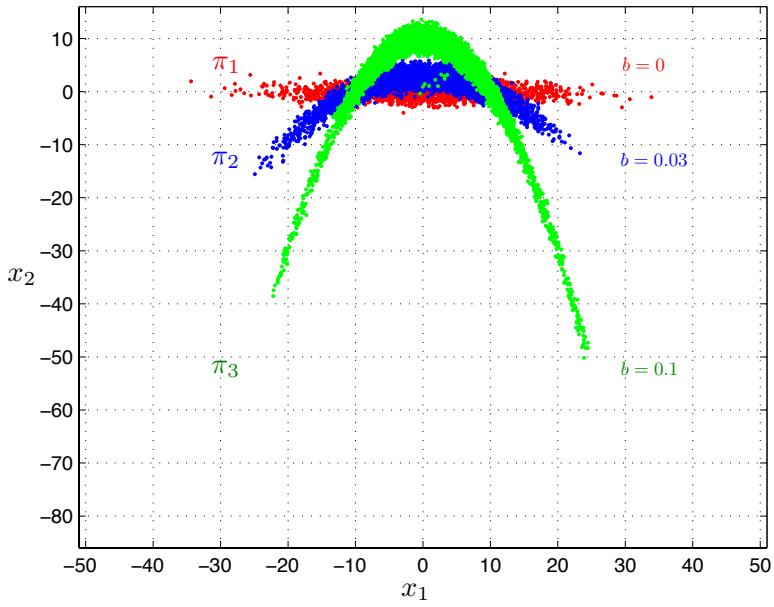



Figure 5.2: Complete set of M-GaA samples from the test target distributions $\pi_1$ (red), $\pi_2$ (blue), and $\pi_3$ (green) for one run, randomly selected from the 100 runs. A 2D projection of the 8-dimensional data set is shown.

The performance measures for all algorithms are summarized in Table 5.1. All results other than those for M-GaA are taken from (Haario et al., 1999). For the uncorrelated target $\pi_1$, M-GaA and AP both outperform MH in estimating the expectation value. They also show a lower standard deviation of the estimation. While the samples from M-GaA have a bias of around 4% in the 68.3% region, they accurately cover the tails beyond the 99% region. For the twisted Gaussians $\pi_2$ and $\pi_3$, M-GaA and AP estimate the expectation more accurately than SC and MH. This indicates that M-GaA is able to better explore the twisted tails of the distributions, leading to a smaller error in the expectation estimation. An important feature is that for all twisted distributions the M-GaA estimates have smaller standard deviations than those from any of the other algorithms. This is an appealing property in practice.

We study the mixing behavior of the algorithms by computing the component-wise autocorrelation $\rho$ of the M-GaA and MH samples (Fig. 5.3). For $\pi_1$, the sample components $x_1$ (along the stretched axis) are much less correlated in M-GaA than they are in MH. The same is true for $\pi_2$ in both the first and second dimension (stretched and twisted). All other components show low correlations in both algorithms, with the MH sample autocorrelation dropping slightly faster than that of M-GaA (curves in the lower-left corner of the graphs in Fig. 5.3). We observe the same behavior also for the target $\pi_3$ (data not shown). Altogether, the strong reduction in sample autocorrelation indicates fast mixing of the chains produced by M-GaA.

| $\pi_1$ | | | | |
|---|---|---|---|---|
| | **SC** | **MH** | **AP** | **M-GaA** |
| **mean**($\|E\|$) | - | 2.96 | 0.46 | 0.62 |
| **std**($\|E\|$) | - | 2.31 | 0.33 | 0.44 |
| **err**($\leq 68.3\%$) | - | 0.23 | 0.02 | 4.29 |
| **std**($\leq 68.3\%$) | - | 4.40 | 1.95 | 2.41 |
| **err**($> 99\%$) | - | 0.01 | 0.03 | 0.04 |
| **std**($> 99\%$) | - | 0.61 | 1.32 | 0.39 |
| $\pi_2$ | | | | |
| | **SC** | **MH** | **AP** | **M-GaA** |
| **mean**($\|E\|$) | 2.40 | 2.46 | 1.31 | 1.48 |
| **std**($\|E\|$) | 4.59 | 2.81 | 0.72 | 0.71 |
| **err**($\leq 68.3\%$) | 1.30 | 0.18 | 0.80 | 0.29 |
| **std**($\leq 68.3\%$) | 4.59 | 6.70 | 2.92 | 1.95 |
| **err**($> 99\%$) | 0.16 | 0.03 | 0.01 | 0.16 |
| **std**($> 99\%$) | 0.40 | 0.66 | 0.62 | 0.25 |
| $\pi_3$ | | | | |
| | **SC** | **MH** | **AP** | **M-GaA** |
| **mean**($\|E\|$) | 6.53 | 7.89 | 4.85 | 4.96 |
| **std**($\|E\|$) | 4.79 | 7.54 | 4.20 | 1.14 |
| **err**($\leq 68.3\%$) | 2.46 | 0.35 | 2.13 | 1.27 |
| **std**($\leq 68.3\%$) | 6.48 | 9.79 | 5.34 | 2.56 |
| **err**($> 99\%$) | 0.27 | 0.07 | 0.14 | 0.26 |
| **std**($> 99\%$) | 0.34 | 0.97 | 0.45 | 0.28 |

Table 5.1: Summary statistics of 100 independent test runs of the Single-component Metropolis (SC), Metropolis-Hastings (MH), Adaptive Proposal (AP) (taken from (Haario et al., 1999)), and M-GaA samplers. All **err** and **std** values are given in %.

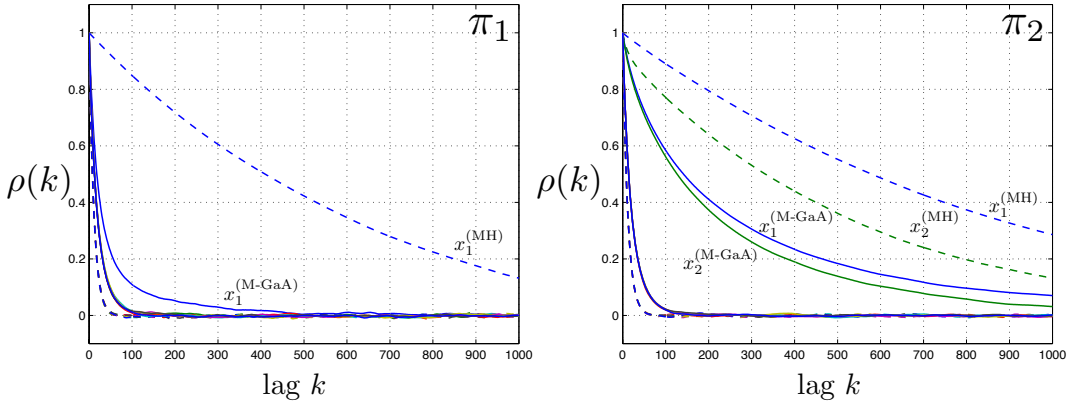



Figure 5.3: Component-wise autocorrelation $\rho(k)$ of the samples (after the burn-in phase) vs. lag $k$ on $\pi_1$ and $\pi_2$ for MH (dashed lines) and M-GaA (solid lines) averaged over 100 runs. The M-GaA samples in the non-standard-normal coordinates ($x_1$ in $\pi_1$ and $x_1, x_2$ in $\pi_2$) are less correlated than the corresponding components in the MH samples. In all standard-normal components, the sample autocorrelation drops fast for both algorithms ($\approx 0$ at $k = 150$).

### 5.3.2 Neal's funnel distribution

In his seminal paper on Slice Sampling (Neal, 2003), Radford Neal introduced a funnel-shaped test distribution $\pi_{\mathrm{f}}(\mathbf{x})$. It is a ten-dimensional distribution with variables $\mathbf{x} = (v, x_1, x_2, \ldots, x_9)$. The marginal distribution of $v$ is $\mathcal{N}(0, 3^2)$. Conditional on a given value of $v$, the variables $x_1$ to $x_9$ are independent, with the conditional distribution for each being Gaussian with $\mathcal{N}(0, e^v)$. The resulting shape of the distribution resembles a single ten-dimensional funnel, with increasing values for $v$ from one end to the other. A contour plot of $\pi_{\mathrm{f}}(\mathbf{x})$ is provided in upper-right part of Fig. 5.4. Neal states that "Such a distribution is typical of priors for components of Bayesian hierarchical models: $x_1$ to $x_9$ might, for example, be random effects for nine subjects, with $v$ being the log of the variance of these random effects. If the data happens to be largely uninformative, the problem of sampling from the posterior will be similar to that of sampling from the prior, so this test is relevant to actual Bayesian inference problems." A direct sampling method is straight forward and consists of first sampling $v^{(g)}$ from $\mathcal{N}(0, 3^2)$ and then, conditionally sampling all $x_i^{(g)}$ from $\mathcal{N}(0, e^{v^{(g)}})$. A representative sample of size $2 \cdot 10^4$, along with the corresponding marginal histograms for $v$ and $x_i$, is depicted in lower and left panels of Fig. 5.4. For standard Metropolis-based schemes, the difficulty of sampling from $\pi_{\mathrm{f}}(\mathbf{x})$ is the low probability of accepting a proposal state when the chain is in a region of negative $v$. Conditional on this value, the variances of $x_i$ attain the tiny value, e.g. 0.018 for v=-4. This leads to highly peaked Gaussians that are difficult to sample from. An MH algorithm with a standard multivariate Gaussian proposal $\mathcal{N}(\mathbf{0}, \mathbf{I})$ would, in the majority of cases, propose samples that are rejected and the chain would get stuck on the lower end of the funnel. The same also hinders the chain from visiting negative $v$ values when started from positive values, because these moves are almost always rejected. Monitoring the $v^{(g)}$ values of the chain reveals, hence, the efficiency of an MCMC method.

We demonstrate the limitations of M-GaA using similar numerical experiments as in Neal's article. Following Neal, we run a single long trajectory for each sampler. Both for N-SRWM version of the MH algorithm and for standard M-GaA we use $2 \cdot 10^5$ iterations. We thin out the chain and record only every tenth sample point $j = 1, \ldots, 2 \cdot 10^4$ for analysis. In addition, we run a single trajectory of Neal's slice sampler of length $2 \cdot 10^4$ samples (using MATLAB's *slicesample.m* routine with standard settings). M-GaA is run with standard parameters, except for a more conservative value of $N_{\mathrm{C}} = 5\,n^3$ for the covariance update. The initial state of all chains is set to $\mathbf{x}^{(0)} = [0, 1, 1, 1, 1, 1, 1, 1, 1, 1]$. We first analyze the trace of variable $v$ as summarized in Fig. 5.5. The true marginal distribution of $v$ is the Gaussian distribution $\mathcal{N}(0, 3^2)$. Visual inspection of the histograms (lower row of Fig. 5.5) suggests that M-GaA is superior to the other two methods. Both MH and Slice sampling get stuck for small values of $v$ and produce an additional mode in the region of $v \approx -10$. The empirical estimates of the first two moments of the M-GaA samples are $m_{\text{M-GaA}} = 0.04$ and $var_{\text{M-GaA}} = 3.38^2$, whereas for MH and the Slice sampler we have $m_{\mathrm{MH}} = -0.35$, $var_{\mathrm{MH}} = 4.86^2$ and $m_{\mathrm{Slice}} = -1.10$ and $var_{\mathrm{Slice}} = 5.95^2$, respectively.

Inspection of the marginals of the $x_i$, however, reveals that the M-GaA trajectory diverges (in terms of variance) over time while both the MH and Slice Sampler are stable. For illustration, we show the trajectory of $x_1$ in Fig. 5.6. We hypothesize that the maximum

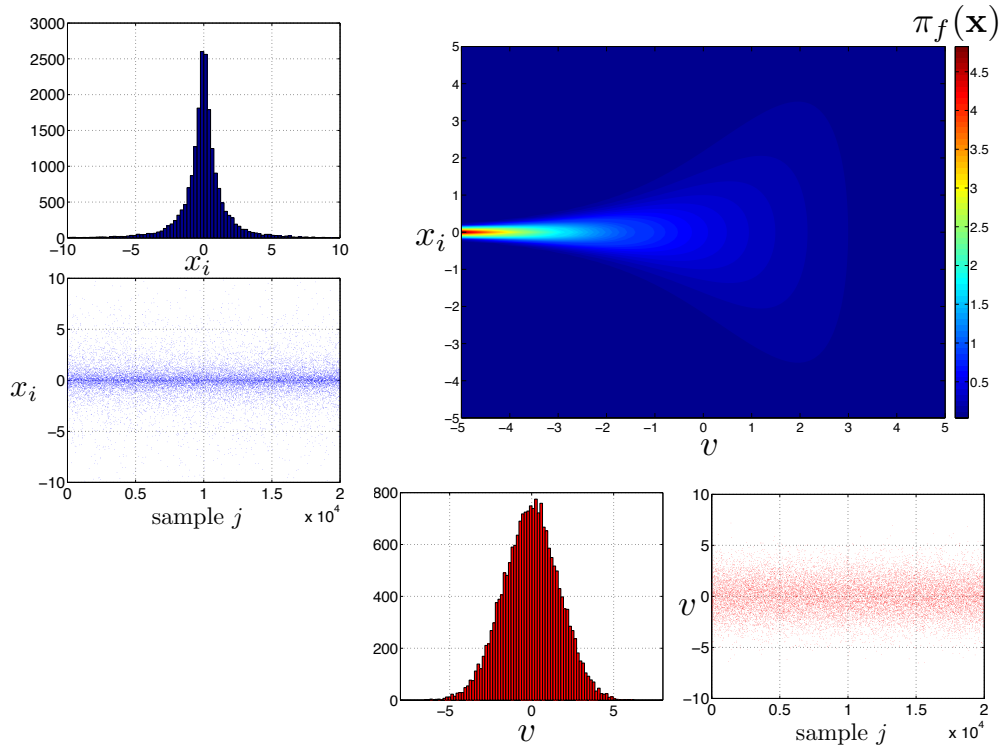



Figure 5.4: Top-right panel: Contour plot of the density $\pi_f(\mathbf{x})$ as a function of $v$ and an arbitrary $x_i$ in the domain [-5,5]. Direct samples from this distributions for $v$ (red) and an arbitrary $x_i$ (blue) are shown in the lower right and upper left panels, respectively.

entropy adaptation in M-GaA is responsible for this divergence. When M-GaA explores the target distribution for large values of $v$, the distribution is almost uniformly flat in the $x_i$ components (due to the exponential variance $e^v$ of these variables). M-GaA thus expands the covariance of the Gaussian proposal along these directions. When the trajectory of M-GaA returns to negative $v$, the sampler is blind to the peak in the target distribution at 0 in the $x_i$ dimensions. M-GaA rather explores the uninformative flat regions away from 0 and further expands the covariance matrix. This leads to the divergence of the chain in the $x_i$ components.

The divergence of the variables $x_i$ clearly demonstrates the limitations of the current M-GaA sampler. Due to the hierarchical dependency of the $x_i$ variables on $v$ in the funnel distribution, it is hard to learn an optimal scale and direction of the covariance matrix in all regions of the domain. Several simple modifications are conceivable for M-GaA. First and foremost, the principle of vanishing adaptation certainly helps to avoid the divergence of the proposal distribution. Second, practical bounds can be enforced on the domain of interest and on the condition number of the covariance matrix. These are, in essence, the same strategies that are

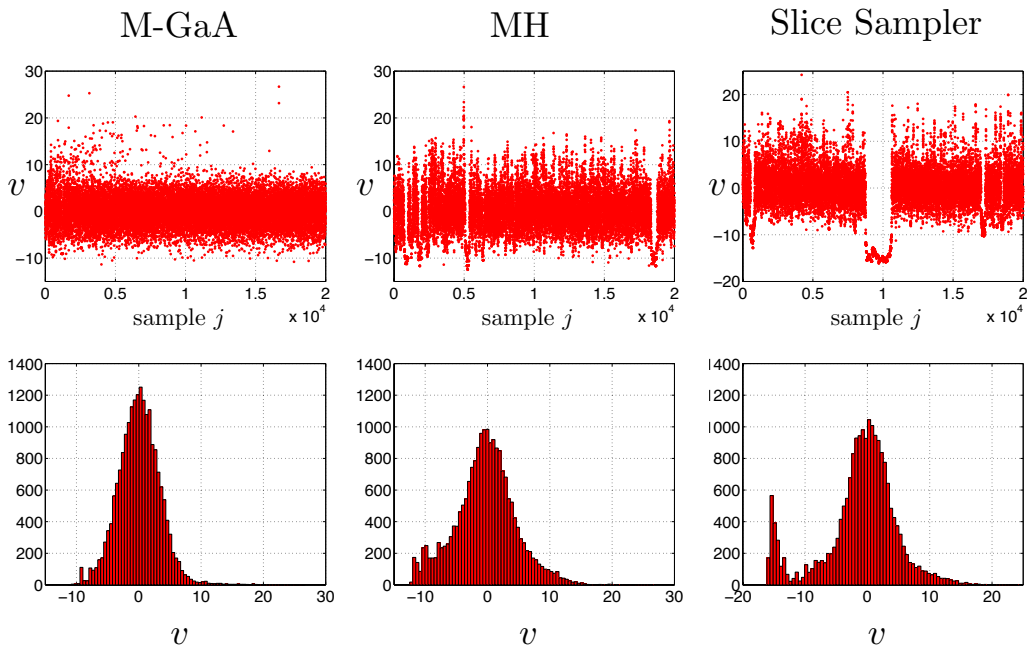



Figure 5.5: Markov chain samples from $\pi_f(\mathbf{x})$ obtained using M-GaA (left), the MH algorithm (middle) and the Slice Sampler (right). Both the full trajectories of $v$ (top) and the corresponding histograms (bottom) are shown.

also used in the AM algorithm.

## 5.4 A future challenge: A unifying framework for black-box optimization and sampling

The previous Chapter 4 and this chapter explored adaptive algorithms for black-box optimization and sampling. In identifying and analyzing the common design principles and features of CMA-ES, GaA and adaptive MCMC methods, we have been able to synthesize a novel adaptive MCMC method, M-GaA, that shows encouraging performance on the presented test problems. In our view, this is just a first step toward a unifying framework for adaptive black-box optimization and sampling. Exploring the key ideas from both communities and adapting them into the respective context is expected to be of mutual advantage. Besides our own work, also other attempts have been made in this direction. In statistics, Liang and Wong introduced an Evolutionary Monte Carlo scheme, an MCMC method that uses move

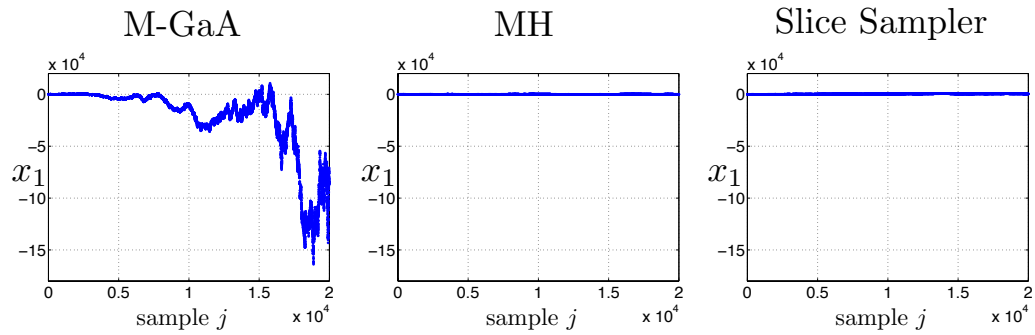



Figure 5.6: Markov chain samples of $x_1$ from $\pi_f(\mathbf{x})$ obtained using M-GaA (left), the MH algorithm (middle), and the Slice Sampler (right).The M-GaA trajectory diverges in time. The MH and the Slice Sampler are stable.

sets from evolutionary algorithms (Liang and Wong, 2001b,a). Several authors have recently refined such schemes (Braak, 2006; Hu and Tsui, 2008). A combined approach to sampling and optimization is proposed in (Ren et al., 2008). In the black-box optimization community, few authors have recognized the possible importance of MCMC methods. An important exception are the works of Vrugt and co-workers. For instance in (Vrugt and Robinson, 2007), the AM algorithm is used as part of an ensemble of search/sampling methods to improve multi-objective optimization.

We expect research into MCMC methods to be promising for the solution of two important problems in black-box optimization: (i) Efficient high-dimensional versions of CMA schemes and (ii) convergence proofs of CMA schemes for convex and non-convex landscapes.

The first challenge can be tackled by learning from the AP algorithm's sampling scheme as given in Eq. (5.14). The idea of using a $H \times n$ data matrix of past accepted samples that is used as an approximation of the Cholesky matrix may prove valuable also in the optimization context. For high-dimensional problems ($n > 500$), methods that adapt a full covariance matrix are largely prohibitive due to the cubic complexity of the necessary eigendecomposition. In the AP scheme, this is avoided at the expense of a larger matrix-vector multiplication when $H > n$. The cost of such a multiplication is, however, negligible compared to a Cholesky- or eigendecomposition. To some extent, this idea is reminiscent of the generating set adaptation scheme (Hansen et al., 1995), a precursor of CMA-ES. We suggest revisiting this approach for high-dimensional cases although considerable doubts about its usefulness have been expressed by one of the present co-examiners (Hansen, 2010a). An alternative approach toward high-dimensional CMA versions is offered by updating structured covariance matrices rather than the full covariance. While the single-component case has been introduced by Haario and co-workers for the AM algorithm in (Haario et al., 2005), Ros and Hansen proposed a similar scheme for CMA-ES in (Ros, Raymond and Hansen, Nikolaus, 2008) by updating a diagonal covariance matrix. It is conceivable that this idea can also be extended to block-wise updates of covariance matrices (Andrieu and Thoms, 2008) when certain variables are known or suspected

to be independent.
The second challenge is theoretically more profound. Since Haario's proof of the ergodicity of the AM algorithm, statisticians have worked on simplified proofs or general conditions on adaptive MCMC methods that imply convergence (Andrieu and Moulines, 2006). We strongly believe that these results can be exploited in order to find proofs of convergence to the global minimum for CMA-like schemes on specific model landscapes.

# 6

# Atomic Cluster Landscapes for Black-box Optimization

> "Nucular, it's pronounced 'nucular'...'nucular'"
> Homer Simpson, in: The Simpsons, Bart Gets an Elephant, Episode no. 98, 1999

The energy landscape perspective has been key for the understanding of physical properties of atomic systems. In Chapter 2 we presented several instances where the topology of the potential energy landscape (PEL) determines both static and dynamic properties of atomic ensembles. Depending on the physical properties of the nuclei (and electrons), the continuous landscape arising from their interactions enables rapid formation of regular structures, such as crystals, or unstructured arrangements usually referred to as glasses (Cox et al., 2006). Over the past six decades chemists and physicists have accumulated a wealth of knowledge about a variety of systems. For an exhaustive summary of this field we refer to (Wales, 2005). Several instances have also attracted the attention of mathematicians and computer scientists in the field of discrete and *computational geometry*. The problem of finding an arrangement of particles that minimizes a potential energy is a specific instance of a *geometry optimization* or *packing problem*. These problems generally belong to the class of NP problems.

We here propose the energy minimization of atomic clusters as a promising problem class for continuous black-box optimization benchmarks. From the large set of available cluster optimization problems, we focus on two specific instances: Cohn-Kumar clusters and Lennard-Jones clusters. The potential energy of these clusters is governed by distance-dependent pairwise interaction potentials. The resulting landscapes exhibit smooth and rugged single-funnel

topologies as well as s*tunable* double-funnel topologies. We argue that minimizing the energy landscapes of atomic clusters provides a useful extension to the current CEC and BBOB test sets for two main reasons: First, these problems comprise the property of *isospectral symmetry*, a characteristic that is not covered by the current benchmark sets. Second, atomic cluster problems can be considered real-world problems since they model physical phenomena and share a similar problem structure with other important real-world optimization tasks, such as sensor placement problems (Wu and Verma, 2008). Algorithms that perform well on atomic cluster problems may therefore also prove successful for these applications. We therefore suggest that the presented problem instances should be included in future black-box benchmark suites.

## 6.1 Cluster landscapes

A cluster is a spatial arrangement of particles, typically a few tens to hundreds in number. In chemistry and physics, the study of clusters of atoms provides a means of understanding nucleation phenomena. Nucleation describes the transition from a loose collection of atoms to a bulk material with particle numbers on the order of the Avogadro constant $N_\mathrm{A} = 6.02210^{23}$. Depending on chemical composition and physical conditions, different types of solids, such as crystals, quasi-crystals, amorphous solids, or glasses emerge. The energy landscape paradigm has provided a fruitful view of these complex processes. It is now widely acknowledged that the global topology of a PEL has an important influence on the way atomic clusters and bulk materials form. When the PEL has a funnel-like shape where local minima are arranged in order of decreasing energy around the global minimum (or ground state), rapid evolution of the physical system toward this ground state is likely. Understanding and elucidating energy landscapes both in computer models and real experiments is thus an important research goal. In theoretical and computational approaches, PEL can be discriminated by three fundamental model assumptions about the underlying physical system: (i) the number of particles in the system; (ii) the number of different atom types, and (iii) the classical or quantum-mechanical formulation of the energy. We first discuss the implications of these different model assumptions and state the choices we made in order to derive feasible benchmark problems for black-box optimization.

### *Atomic ensembles and landscape domains*

The number of atoms in the system is the first important choice. Bulk systems with a huge number of atoms are approximated by a finite number of $N$ atoms located in a rectangular prism (or a general parallelepiped) with periodic boundary conditions. In chemistry and crystallography, this rectangular prism is usually called the unit cell. When different kinds of atoms are considered, the unit cell should reflect the stoichiometry of the material, that is, the absolute ratio of the different atom types in the system. The correct size of the unit cell and the number of atoms it contains are usually not known *a priori*, thus hampering the setup of generic problem classes. The landscape domain $\mathcal{X}$ of the described systems is determined by the number of atoms, the spatial extent of the unit cell, and a finite set of indicator variables

that encode the atom types.

When the number of atoms is limited (usually to a few tens or hundreds), the spatial location of the atoms is bounded and no periodic boundary conditions are applied. This leads to *atomic cluster* or *nano cluster* systems. Over the past three decades, there has been increasing interest in such systems because of their wide-ranging chemical applications in catalysis, electronics, and energy conversion. We refer to (Catlow et al., 2010) for an excellent chemical perspective on the topic. Many cluster systems have been studied, and the putative ground states (i.e., configurations of atoms attaining the global energy minimum) for different energy formulations are available, for instance in the Cambridge Cluster Database (CCD) (Wales et al., 2009). Cluster systems comprising mixtures of one, two, or three different atoms types are well studied, most prominently oxides such as Zinc oxides or Silica. Two different definitions of the space in which atomic clusters live (i.e., their *ambient space*) are commonly used: Either the atoms populate the box-constrained Euclidian space, or they are restricted to the surface of the unit sphere $\mathcal{S}^{n-1} = \{\mathbf{x} \in \mathbb{R}^n : d_{\mathrm{E}}(\mathbf{0}, \mathbf{x}) = 1\}$. For our benchmark problems, we consider both spaces. Finding particle distributions on the unit sphere that minimize a certain energy function is a long-standing problem since Thomson posed the question how to optimally arrange electric charges on a sphere (Thomson, 1904). Optimal distributions of particles on the sphere are also known as *spherical codes* in coding theory. A collection of putative optimal spherical codes can be found in (Sloane et al., 2000). Optimality is always defined with respect to a potential energy formulation.

### *Potential energy formulations and ground states*

When a collection of particles interacts on the atomic scale, the potential energy landscape arises from the forces between all electrons and nuclei of the atoms. In order to arrive at a tractable model of the PEL, the fundamental assumption in both classical and quantum-mechanical formulations is the *Born-Oppenheimer approximation* (Born and Oppenheimer, 1927): Based on the large discrepancy between nuclear and electronic masses, this approximation allows separating the energy into an electronic and a nuclear component. Existing formulations of potential energy differ in the way they describe these components. Depending on the specific properties of the atoms in the system, different levels of detail are necessary in realistic models. Two main models gained popularity in the past decades: quantum mechanical (QM) *electronic structure techniques* and classical *interatomic potentials*. Despite today's increasing availability of high-performance computing environments, QM-based energy calculations of atomic ensembles are still very costly. Classical interatomic potentials offer a convenient alternative to calculate the potential energy of many-particle systems in a fast manner. The principle underlying the design of these potentials is largely empirical. One attempts to reproduce the experimentally observed dynamics and ground states of specific types of matter through careful parametrization of simple analytical functions. Consider a cluster of $N$ atoms in 3D space where the position of the $i^{\mathrm{th}}$ particle is denoted $\mathbf{p}_i = (x_i, y_i, z_i)$. Each configuration $\mathbf{x} = \{\mathbf{p}_1, \ldots, \mathbf{p}_i, \ldots, \mathbf{p}_N\}$ is restricted to a 3D box, i.e. $\mathbf{x} \in \mathcal{X} = [\mathbf{l}, \mathbf{u}] \subset \mathbb{R}^{3N}$.

The general form of an interatomic potential energy $E_{\mathrm{IP}}$ for many-body systems is:

$$E_{\mathrm{IP}}(\mathbf{x}) = \sum_{\mathbf{p}_i} v_{\mathrm{IP}}\left(\mathbf{p}_i, \mathcal{N}(\mathbf{p}_i)\right) , \tag{6.1}$$

where $\mathcal{N}(\mathbf{p}_i)$ represents the neighborhood of $\mathbf{p}_i$. An established approach for designing interatomic potentials is to split $v_{\mathrm{IP}}$ into two components. One component accounts for many-body energy contributions through pairwise interactions. The other component models the local environment of each individual atom. In general, the pairwise interaction term depends on all particles in the system, whereas the local environment is often defined within a specified distance range (neighborhood). Important examples of interatomic potentials that include both terms are the Finnis-Sinclair potential (Finnis and Sinclair, 1984) and its extension, the Sutton-Chen potential (Sutton and Chen, 1990), as well as tight-binding potentials (Cleri and Rosato, 1993). The famous Stillinger-Weber potential includes 2- and 3-body interactions (Weber and Stillinger, 1985). The simplest instances of interatomic potentials are those that only consider pairwise interactions using *isotropic* pair potentials. The first potential of this kind dates back to John Lennard-Jones who introduced an empirical potential that describes the interaction between neutral atoms (Lennard-Jones, 1924). This Lennard-Jones (LJ) potential will be considered in Section 6.4. Another important instance is the Morse pair potential (Morse, 1929). Although all presented interaction potentials can be parameterized for different atom types, and atomic mixtures, we only consider mono-atomic clusters, i.e., clusters that only comprise one atom type. The Thomson problem mentioned above can also be considered a minimization problem over a potential energy of pairwise interactions. There, the $N$ particles are electrons confined to the sphere that are arranged such as to minimize the total Coulomb potential.

The quality of potential energy models is usually assessed by their ability to either reproduce experimentally known ground states for various materials and clusters or to *predict* novel geometries as possible ground states that could guide experimentalists in their quest for novel forms of matter. Ground states define the macroscopic properties of the material. In a series of articles, Rechtsman, Stillinger, and Torquato (Rechtsman et al., 2005, 2006a,b) introduced a new perspective on the topic: Instead of attempting to mimic nature as accurately as possible by tuning potential functions, they considered the inverse problem of how the shape of an isotropic pair potential has to look like in order to have as a ground state a predefined structure. In their simulation-guided optimization framework they were able to design interaction potentials that result in bulk material with honeycomb (Rechtsman et al., 2005), square (Rechtsman et al., 2006a), cubic (Rechtsman et al., 2006b), diamond, and wurtzite (Rechtsman et al., 2007) lattices. All of their designed interaction potentials are based on multi-modal *isotropic* pair potentials. Inspired by this "inverse statistical mechanics" approach (Torquato, 2009), Cohn and Kumar derived in their article "Algorithmic design of self-assembling structures" (Cohn and Kumar, 2009) convex pair potentials with *provable* ground states for clusters. We introduce two of these important potentials in Section 6.3 and propose the resulting PEL as a benchmark problem with a smooth single-funnel topology.

### *Standard identification of ground states*

Atomic clusters and their ground states constitute an important problem class in chemistry and physics. From a computational point of view, it is noteworthy that Wille and Vennik proved "that determining the ground state of a cluster of identical atoms, interacting under two-body central forces, belongs to the class of NP-hard problems" (Wille and Vennik, 1985). The proof works by reducing the problem to a special instance of the traveling salesman problem. Recently, Adib revisited and refined the proof (Adib, 2005). Because of the existence of analytic gradients in most potential energy formulations, competitive algorithms to identify the ground states of larger clusters are hybrid stochastic-deterministic first-order methods that give no guarantees about the quality of the found solutions. Despite the tremendous amount of different approaches to cluster optimization found in the (mostly chemistry or physics) literature, the basic ingredients of successful heuristics are few: a reasonably good *global* move set and an efficient Quasi-Newton *local* gradient-based minimizer, mostly of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) type. We refer to Wales's book (Wales, 2005) for further details, and to the excellent review by Hartke for an overview of applications of hybrid evolutionary algorithms (Hartke, 2004). The memetic CMA-ES algorithm presented in Section 4.2.2 is an instance of such a heuristic that has been applied to LJ clusters (Ofenbeck, 2009). One of the most successful hybrid algorithms is the Basin-Hopping (BH) algorithm (Wales and Doye, 1997), which is based on (i) uniformly random variation of atomic positions and (ii) relaxation of the perturbed structure to the nearest local minimum using BFGS. In this context, a "basin" is the collection of configurations that lead to the same local minimum for a given gradient-based minimization routine. Acceptance of a new structure is based on the Metropolis criterion, as defined in Eq. (5.9), with respect to the previously accepted configuration. This is analogous to Simulated Annealing. In BH, however, the temperature parameter remains constant. Numerical results of BH and related memetic techniques for different cluster systems are scattered over hundreds of publications. For some systems, detailed knowledge about the number of minima, first-order saddle points, and global landscape structure is available. The present investigation is a first attempt to utilize this information for black-box benchmarking.

## 6.2 Cluster problems for black-box benchmarking

The physical and computational foundations of atomic clusters can be exploited for the development of a cluster benchmark library. We adopt several design criteria of the CEC 2005 test suites. The dimensionality $n$ of a cluster problem instance depends on the number of particles in the cluster and the space that the individual atoms populate. We consider problems not exceeding $n \approx 100$. Furthermore, we restrict ourselves to clusters of identical atoms with energy formulations based on scalar, isotropic pair potentials $u_{\mathrm{PP}}$. This leads to an energy formulation of the kind:

$$E_{\mathrm{PP}}(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} u_{\mathrm{PP}}(r_{ij}) \,, \tag{6.2}$$

where $r_{ij} = d_{\mathrm{E}}(\mathbf{p}_i, \mathbf{p}_j)$ is the Euclidian distance between particles $\mathbf{p}_i$ and $\mathbf{p}_j$. The sole dependence of the isotropic pair potential on pairwise distances has several consequences: First,

the resulting energy functions are non-separable because all particles interact with all others. Second, they are, in general, continuously differentiable polynomial surfaces that are not bounded from above because most pair potentials diverge when $\mathbf{p}_i$ approaches any $\mathbf{p}_j$. Third, the computational cost of energy function evaluation is low, but scales quadratically with the number of particles (and hence dimensions). Forth, there is no unique set of absolute atomic positions that minimizes the energy; rather, all configurations that have the same distance spectrum are energetically equivalent. This property is called *isospectral symmetry* and will be discussed in detail in the next subsection. It is the key feature of cluster problems that is not represented in current standard benchmark suites.

A convenient property of atomic cluster problems is the intuitive illustration of (sub-)optimal solutions as a list of atomic coordinates that can be visualized in 3D space. In addition, the physical concept of *order parameters* enables low-dimensional descriptions of cluster configurations, as will be outlined below. Cluster problems also exhibit a wide variety of landscape topologies, thus leading to a rich and diverse benchmark set in the spirit of the CEC2005 or BBOB test suites.

### *Symmetry as a novel problem characteristic*

Two sources of symmetry arise in mono-atomic clusters: Consider the global minimum $\mathbf{x}^*$ for a given energy function. When $\mathbf{x}^*$ consists of the positions of $N$ identical particles, $N!$ possible permutations of atomic positions exist that attain the same ground state. Furthermore, the energy function does not discriminate between configurations with identical sets of pairwise distances, that is, an identical *distance spectrum*. This characteristic, known as *isospectral symmetry*, implies that any transformation applied to $\mathbf{x}^*$ that preserves all pairwise distances between the particles results in another minimum-energy configuration. A unique description of the cluster ground state is thus a set of $N(N-1)/2$ pairwise distances, rather than a set of coordinates $\mathbf{x}$. Depending on the symmetry group of the minimum-energy configuration, different types of spectrum-preserving transformations exist. The simplest ones are translation, rotation, inversion, and reflection. However, it is also known that certain distance spectra can be generated by geometrically distinct configurations. Note that this type of symmetry is conceptually different from the symmetries found in standard benchmark functions, such as Rastrigin's function, where permutation symmetries often exist for *local* minima, but no symmetry exists for the global minimum. In the BBOB and CEC 2005 test suites symmetries are also partially removed by (non-)linear transformations (Suganthan et al., 2005; Finck et al., 2009). Isospectral symmetry (and symmetry breaking) is a fundamental concept in physics, but its impact on black-box optimization has been comparatively weak. We are only aware of the research of Van Hoyweghen and co-workers who analyzed symmetry "due to the interaction structure" in the problem. They describe the impact of such symmetry on the performance of evolutionary algorithms in the context of aggregated combinatorial problems (Van Hoyweghen and Naudts, 2000; Van Hoyweghen et al., 2002). They "claim that the occurrence of symmetry in the representation is another problem difficulty characteristic" (Van Hoyweghen et al., 2002), and they propose different ways of modifying black-box algorithms in order to cope with this difficulty. For continuous black-box problems such considerations are, to the best of our knowledge, so far missing.

### *Bond-order parameters for cluster characterization*

Due to the presence of isospectral symmetry, it can be misleading to describe and compare cluster configurations in absolute coordinates $\mathbf{x}$. Consider two configurations $\mathbf{x}$ and $\mathbf{y}$, where $\mathbf{y}$ is generated by rotation of $\mathbf{x}$ about the origin. Calculating $d_{\mathrm{E}}(\mathbf{x}, \mathbf{y})$ would result in a value greater than 0, despite the fact that the two configurations are identical in terms of their pairwise distance spectrum. Measures that are invariant to isospectral symmetries provide a more robust way of characterizing the system. Designing such invariants for specific systems, however, is not trivial. Steinhardt and coworkers introduced a set of invariants for atomic cluster and bulk configurations that are applicable in our case: *bond-(orientational) order parameters* (BOP) (Steinhardt et al., 1983). These parameters are indispensable for the analysis of nucleation phenomena and packing structures in bulk materials and cluster configurations. In this context, a "bond" does not refer to a covalent chemical bond, but is rather defined as the vector joining a pair of neighboring atoms. Neighborhood is defined by a distance cutoff. Bond-order parameters reflect the symmetry of bond orientations, *regardless* of absolute bond lengths. This is achieved by combination and normalization of certain spherical harmonics, resulting in the measures $Q_l$ and $\hat{W}_l$. The second-order invariants $Q_l$ are defined as:

$$Q_l = \left( \frac{4\pi}{2l+1} \sum_{m=-l}^{l} \|\bar{Q}_{lm}\|^2 \right)^{\frac{1}{2}} , \tag{6.3}$$

where

$$\bar{Q}_{lm} = \frac{1}{N_b} \sum_{r_{ij}<r_0} Q_{lm}(\mathbf{r}_{ij}) \tag{6.4}$$

and $Q_{lm}(\mathbf{r}_{ij}) = Y_{lm}(\theta_{ij}, \phi_{ij})$. $N_b$ denotes the number of bonds that are shorter than the cutoff distance $r_0$. The $Y_{lm}(\theta_{ij}, \phi_{ij})$ are spherical harmonics with $\theta_{ij}$ being the polar and $\phi_{ij}$ the azimuthal angle of the inter-atomic vector $\mathbf{r}_{ij}$ of length $r_{ij}$ between atoms $\mathbf{p}_i$ and $\mathbf{p}_j$ with respect to an arbitrary coordinate frame. The parameters $\hat{W}_l$ are defined as:

$$\hat{W}_l = \frac{W_l}{\left( \sum_m \|Q_{lm}\|^2 \right)^{3/2}} . \tag{6.5}$$

They are normalized versions of the third-order invariants

$$W_l = \sum_{\substack{m_1,m_2,m_3 \\ m_1+m_2+m_3=0}} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} \bar{Q}_{lm_1} \bar{Q}_{lm_2} \bar{Q}_{lm_3} , \tag{6.6}$$

where the coefficients $(\cdots)$ are the so-called Wigner 3j symbols (Weisstein, 2010). Steinhardt and co-workers showed that the parameters $Q_4$, $Q_6$, $\hat{W}_4$, and $\hat{W}_6$ are sufficient for a detailed "cluster shape spectroscopy" (Steinhardt et al., 1983) of liquids, crystals, and glasses since they discriminate between the most important symmetry groups. Specific atomic packings have unique combinations of values for this set. For instance, the parameter $Q_4$ discriminates between icosahedral (ico) and face-centered cubic octahedral (fcc) packing systems with values

$Q_4^{\text{ico}} = 0$ and $Q_4^{\text{fcc}} = 0.1909$, respectively. In the context of black-box optimization, we suggest using bond-order parameters as structural fingerprints of putative optimal cluster solutions, as well as for convenient visualization of optimization trajectories.

Alternative invariants can be constructed directly from the distance spectrum of the configurations. A wealth of such techniques exists in computer graphics and image processing for symmetry and shape descriptions. One instance is the concept of shape distributions (Osada et al., 2002), which might be interesting to examine also in the context of black-box optimization.

## 6.3 Cohn-Kumar clusters

The first cluster instances we propose are based on pair potentials recently introduced by Cohn and Kumar (Cohn and Kumar, 2009). Inspired by the inverse statistical mechanics approach, they designed pair potentials that result in *provable ground states* for certain cluster instances. To the best of our knowledge, there is so far no publication that uses these potentials. We hence introduce the name "Cohn-Kumar (CK) potentials" for the interaction potentials and "Cohn-Kumar (CK) clusters" for the resulting ground-state clusters. The four pair-potentials (CK1 – CK4) form the following provable minimum-energy configurations:

1. An 8-particle CK1 cluster forms a 3D cube with six identical square faces.
2. A 20-particle CK2 cluster forms a 3D regular dodecahedron with twelve identical pentagonal faces.
3. A 16-particle CK3 cluster forms a 4D hypercube with eight identical *cubic* faces.
4. A 600-particle CK4 cluster forms a regular 120-cell in 4D with 120 dodecahedral faces.

We restrict ourselves to the first two potentials with ground-state clusters living in 3D space. In both cases, the ambient space of the particles is $\mathcal{S}^2$, leading to two degrees of freedom per particle. The CK1 pair potential is defined as:

$$u_{\text{CK1}}(r) = \frac{1}{r^3} - \frac{1.13}{r^6} + \frac{0.523}{r^9}\,, \tag{6.7}$$

where $r$ is the Euclidian distance between two particles. The CK2 pair potential is defined as:

$$u_{\text{CK2}}(r) = (1+t)^5 + \frac{(t+1)^2(t-1/3)^2(t+1/3)^2(t^2-5/9)^2}{6(t-1)^2} \tag{6.8}$$

with $t = 1 - r^2/2$. Both functions are designed to be monotonically decreasing and convex. Their graphs are depicted in Fig. 6.1. For a system of $N$ particles with positions $\mathbf{p}_i$, the energy functions are:

$$E_{\text{CK1}}(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} u_{\text{CK1}}(r_{ij})\,, \tag{6.9}$$

and

$$E_{\text{CK2}}(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} u_{\text{CK2}}(r_{ij})\,, \tag{6.10}$$

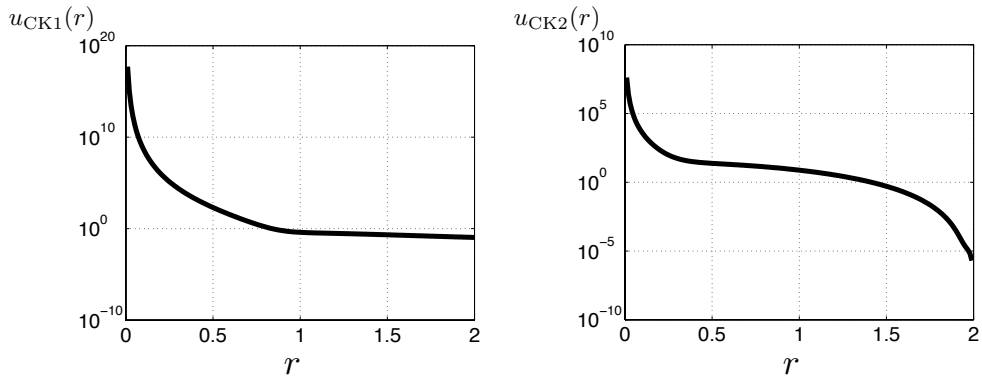



Figure 6.1: The Cohn-Kumar (CK) potentials $u_{\mathrm{CK1}}(r)$ and $u_{\mathrm{CK2}}(r)$ in log scale versus distance $r \in [0.01, 2]$.

with $\mathbf{x} = \{\mathbf{p}_1, \ldots, \mathbf{p}_i, \ldots, \mathbf{p}_N\}$ and $r_{ij} = d_{\mathrm{E}}(\mathbf{p}_i, \mathbf{p}_j)$. Let us first consider the two instances of CK1 and CK2 clusters for which the ground states have been proven in (Cohn and Kumar, 2009). The proofs are based on techniques from coding theory and linear programming. Details are given in (Cohn and Kumar, 2009). The optimal 8-particle CK1 ($\mathrm{CK1}_8$) cluster configuration $\mathbf{x}_{\min}^{\mathrm{CK1}}$ is the 3D cube with identical square faces. This configuration belongs to the class of Platonic solids and is depicted on the left of Fig. 6.2. Due to the restriction of particle positions to the unit sphere, the following 3 unique distances occur: the cube edge length of $2/\sqrt{3}$, the face diagonal of length $\sqrt{8/3}$, and the cube diagonal length of 2. The energy minimum is $E_{\mathrm{CK1}}(\mathbf{x}_{\min}^{\mathrm{CK1}}) \approx 6.34338764$. The dual polygon is the octahedron.

The optimal 20-particle CK2 ($\mathrm{CK2}_{20}$) cluster configuration $\mathbf{x}_{\min}^{\mathrm{CK2}}$ is a dodecahedron with 12 identical pentagonal faces. 5 unique distances occur in this Platonic solid: The length $l_{\mathrm{e}}$ of the pentagonal edges is related to the radius $r_{\mathrm{Sph}}$ of the sphere on which the particles are located via $r_{\mathrm{Sph}} = l_{\mathrm{e}}/4\,(\sqrt{15} + \sqrt{3})$. This leads to $l_{\mathrm{e}} = 4/(\sqrt{15} + \sqrt{3}) \approx 0.713644$. Second, third, and fourth nearest neighbor distances have values $2/\sqrt{3}$, $\sqrt{8}/\sqrt{3}$, and $(\sqrt{15} + \sqrt{3})/3$, respectively. The largest occurring distance is the sphere diameter of 2. The minimum energy is $E_{\mathrm{CK2}}(\mathbf{x}_{\min}^{\mathrm{CK2}}) \approx 746\frac{2}{3}$. The dual polygon is the icosahedron. This dodecahedral configuration is depicted on the right of Fig. 6.2.

Although these configurations are geometrically simple, it is hard to make them ground states of any classical pair-potential energy function. Cohn and Kumar state the following (which is valid for all ground-state configurations listed above): "The problem is that their facets are too large, which makes them highly unstable. Under ordinary potential functions, such as inverse power laws, these configurations are never even local minima, let alone global minima. In the case of the cube, one can typically improve it by rotating two opposite facets so they are no longer aligned. That lowers the energy, and indeed the global minimum appears to be the antiprism arrived at via a $45^o$ rotation (and subsequent adjustment of the edge lengths)." (Cohn and Kumar, 2009). The square antiprism (or anti-cube) is, for instance, the minimum of the 8-electron Thomson problem, as well as the putative ground state for the 8-particle

CK2 cluster.

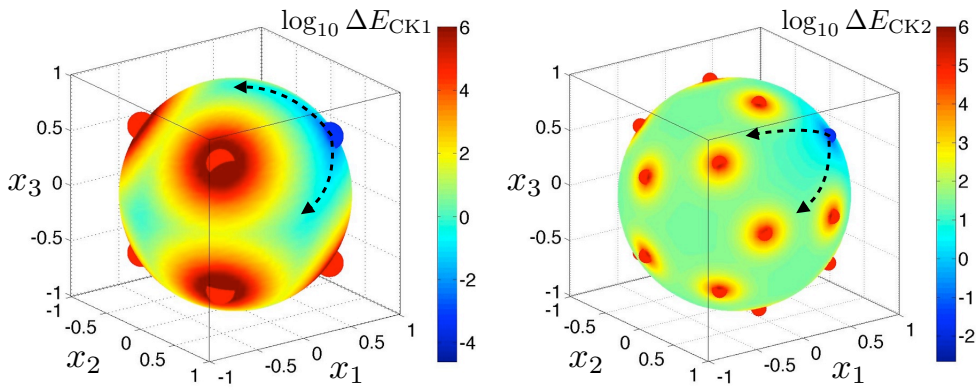



Figure 6.2: The optimal 8-atom CK1 cluster (left) and 20-atom CK2 cluster (right). The color coding represents the log of the energy difference to the ground state of the system if $N-1$ atoms are fixed and a single one (the blue one on the top right) is moved across the surface of the sphere ($\log_{10} \Delta E_{\mathrm{CKi}} = \log_{10} (E_{\mathrm{CKi}}(\mathbf{x}) - E_{\mathrm{CKi}}(\mathbf{x}_{\mathrm{min}}))$), resulting in a multi-funnel landscape with the largest basin surrounding the vacant position.

Despite the convexity of the CK pair potentials, the resulting energy functions are non-convex, even for the clusters with proven optimal configurations. Cohn and Kumar do not comment on the exact number of basins or the basin depths. For the optimal CK1 cluster they empirically found that out of 1000 local optimization runs only 6 did not converge to the global minimum (Cohn and Kumar, 2009). Our own numerical results indicate that the landscapes of the 8-atom CK1 cluster and the 20-atom CK2 cluster are smooth and globally convex under CMA-ES.

These results suggest two ways of using CK clusters as black-box optimization benchmarks: The first one is solely based on the proven optimal configurations. A benchmark set with variable dimensionality can be constructed by fixing a number of optimal atomic positions, leading to well-defined multi-funnel problems. Consider the optimal 8-particle CK1 cluster. Fixing seven particles to optimal relative positions we arrive at a 2D cluster problem with four basins of attraction: three that correspond to the square faces of equal size and one large basin consisting of the remaining three faces with the global minimum at the vacant corner position. Likewise, one could construct a 2D 10-funnel landscape from the optimal 20-particle CK2 cluster with 9 sub-optimal basins (the stable pentagons) and a large basin with the global minimum at the vacant position surrounded by three pentagonal faces. Decreasing the number of fixed atomic positions results in higher-dimensional problems with varying landscape structure. For both provably optimal cluster configurations we illustrate the landscapes where only one atom is free to move in Fig. 6.2 by projecting the energies relative to the ground state onto the spherical ambient space.

The second way of constructing a standard benchmark set for black-box optimization based on CK1 and CK2 clusters consists of varying the number of atoms on the sphere. We prefer this approach for its simplicity and suggest the following specification: We consider CK1 and CK2 clusters with up to $N = 25$ atoms. A system of $N$ atoms confined to the surface of the unit sphere results in $n = 2N$ degrees of freedom. A natural landscape representation is based on spherical coordinates. The position $\mathbf{p}_i$ of atom $i$ is defined by a pair of polar and azimuthal angles at unit radius, i.e., $\mathbf{p}_i = (\theta_i, \varphi_i, 1)$. In order to construct a box-constrained problem we restrict the angles to the interval $[-\pi, \pi]$, leading to $\mathbf{x} \in \mathcal{X} = [-\pi, \pi]^n$. In principle, one could remove 4 degrees of freedom by fixing the position of a single atom on the sphere and fixing one polar and one azimuthal angle of one pair of atoms, thus removing certain symmetry properties of the problem. We do, however, not follow this approach here since we want to construct a benchmark with full isospectral symmetry. As in the CEC 2005 and BBOB test suites, we suggest to use a budget of MAX_FES= $10^4\, n$ for a single optimization run, and 25 independent runs per problem. We first focus on the 8-atom CK1 cluster and the 20-atom CK2 cluster, for which the optimal configurations and the corresponding energy values are known exactly. We then use numerical optimization runs to derive putative global optima for all other instances in Section 6.3. These putative optima are used to specify a termination criterion for the level of solution accuracy $\epsilon_{\text{sol}}$. We also provide BOP values in order to characterize the symmetry of the optimal configurations.

### *Reference black-box experiments on the $CK1_8$ and $CK2_{20}$ cluster problems*

In order to assess the performance of black-box optimizers on the 8-atom CK1 and the 20-atom CK2 cluster problems, we run two sets of numerical optimization experiments, one based on standard IPOP-CMA-ES (with *incPop* = 1.25) (Auger and Hansen, 2005b) and the other based on MATLAB's *fminsearch.m*, a standard tool for black-box search using the Nelder-Mead (NM) simplex method. We follow the above problem specification with $\epsilon_{\text{sol}} = 10^{-6}$ for both problems. Standard parameters for NM and CMA-ES are used, except for the smaller population increase factor for IPOP-CMA-ES. If NM converges before reaching the global minimum, independent restarts are done until the FES budget is exhausted.

The 8-atom CK1 cluster problem poses little challenge to either of the algorithms. Both methods almost always converge to the global minimum within the specified accuracy (CMA-ES with 100% and NM with 96% success rate). The statistics about the minimum energies reached by NM and CMA-ES are summarized in the left part of Table 6.1. The statistics about the number of FES that NM and CMA-ES needed to solve $CK1_8$ are summarized in Table 6.2. From Table 6.2 we see that CMA-ES is more efficient in finding the ground state. While the minimum number of FES is comparable (5317 vs. 5879), the average number of FES (6320.20 vs. 10791.00) is considerably lower for CMA-ES.

The 20-atom CK2 cluster problem reveals a different picture: While CMA-ES always converges to the global minimum within the specified accuracy, Nelder-Mead fails in all 25 runs. The minimum energies reached by NM range from 746.730 to 746.910 (right part of Table 6.1). CMA-ES always reaches the perfect dodecahedral configuration with energy 746.666 without

| **Energy** | **CK1$_8$** | | **CK2$_{20}$** | |
|---|---|---|---|---|
| | **NM** | **CMA-ES** | **NM** | **CMA-ES** |
| **min** | 6.343388 | 6.343388 | 746.730558 | 746.666667 |
| **7th** | 6.343388 | 6.343388 | 746.775331 | 746.666667 |
| **median** | 6.343388 | 6.343388 | 746.825053 | 746.666667 |
| **19th** | 6.343388 | 6.343388 | 746.872171 | 746.666667 |
| **max** | 6.343692 | 6.343388 | 746.910767 | 746.666667 |
| **mean** | 6.343400 | 6.343388 | 746.823867 | 746.666667 |
| **std** | 0.000061 | 0.000000 | 0.052601 | 0.000000 |

Table 6.1: Statistics of the minimum energies reached by NM and CMA-ES for the CK1$_8$ and CK2$_{20}$ cluster problems.

| **Prob.** | | **min** | **7th** | **median** | **19th** | **max** | **mean** | **std** | $p_s$ |
|---|---|---|---|---|---|---|---|---|---|
| **CK1$_8$** | **CMA-ES** | 5317 | 5845 | 6085 | 6745 | 7933 | 6320.20 | 715.43 | 1 |
| | **NM** | 5879 | 7296 | 10791 | 12976 | - | 10298.96 | 2969.71 | 0.96 |
| **CK2$_{20}$** | **CMA-ES** | 36631 | 44086 | 55756 | 66331 | 111736 | 59333.20 | 19133.11 | 1 |
| | **NM** | - | - | - | - | - | - | - | - |

Table 6.2: Statistics of the number of FES used by NM and CMA-ES to reach the optimal configurations for the CK1$_8$ and CK2$_{20}$ cluster problems.

restarts.

Comprehensive structural information about the cluster configurations is provided by Steinhardt's bond-order parameters. In order to compute them, each cluster is augmented by a dummy atom placed at the origin. We suggest using a cutoff distance of $r_0 = 1.01$ for the calculation of BOP values in CK clusters. In Fig. 6.3 we summarize the BOP values for the sub-optimal CK2$_{20}$ configurations found by NM and the optimal ones reached by CMA-ES. We also depict the best configurations found by NM and CMA-ES.

The BOP values for the sub-optimal NM configurations indicate great structural diversity. Although the individual energy values are comparable, the BOP values vary considerably, especially for $Q_6$ and $\hat{W}_6$. Among all BOP traces, the values of the best NM structure (highlighted in red in Fig. 6.3) are the closest ones to the BOP values of a perfect dodecahedron. This is confirmed by visual inspection of this structure, revealing a dodecahedron with distorted pentagonal faces (see right panel of Fig. 6.3), as opposed to most other sub-optimal structures that do not show any clear packing.

The BOP values can also be used to conveniently visualize the optimization path. In Fig. 6.4 we show the trajectories of a typical CMA-ES run on the CK1$_8$ and CK2$_{20}$ cluster problems.

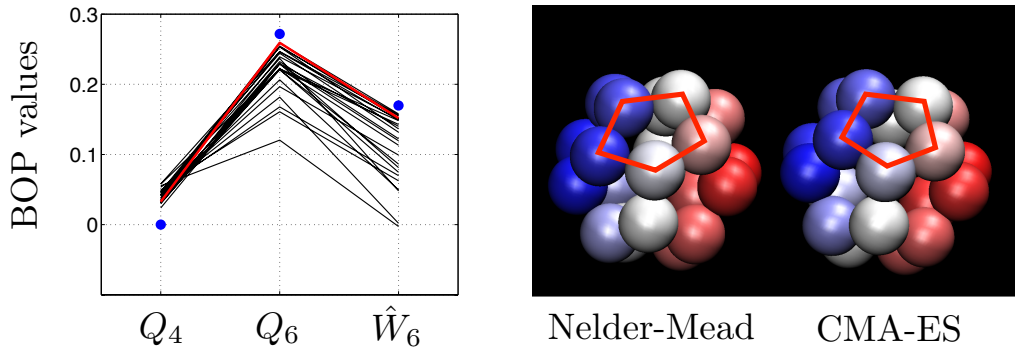



Figure 6.3: The left panel shows BOP values $Q_4$, $Q_6$, and $\hat{W}_6$ for all minimum configurations found by NM (black lines) and CMA-ES (dots). The values for the best NM structure is highlighted in red. The right panel shows this best $CK2_{20}$ configuration of NM and the optimal configuration found by CMA-ES. The NM structure is dodecahedral with distorted pentagonal faces. One face is highlighted in either configuration.

For the $CK1_8$ clusters, the $Q_4$, $Q_6$, $\hat{W}_4$, and $\hat{W}_6$ values of CMA-ES' mean $\mathbf{m}^{(g)}$ converge to

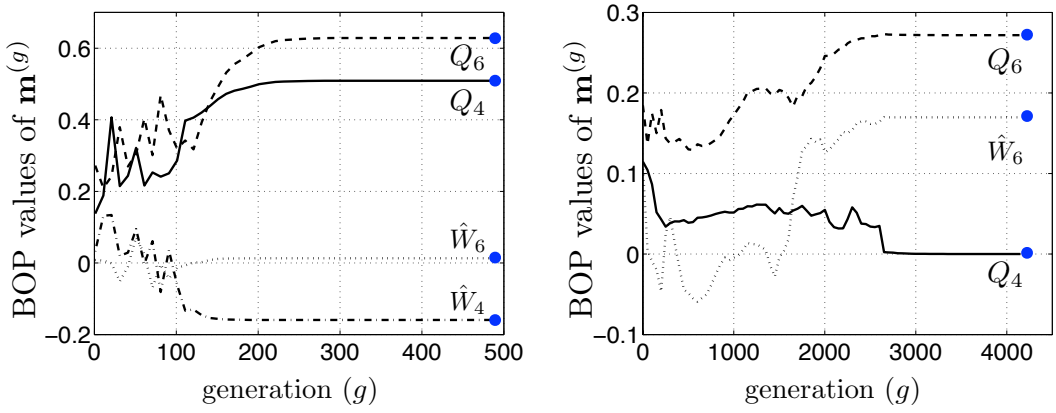



Figure 6.4: Trajectories of typical CMA-ES runs. BOP values of CMA-ES' mean $\mathbf{m}^{(g)}$ are plotted versus generation number ($g$). The left panel shows a CMA-ES trajectory from the $CK1_8$ cluster problem, the right panel one from the $CK2_{20}$ cluster problem. The dots represent the BOP values of the optimal solutions.

the optimal values after about 200 generations. For the $CK2_{20}$ clusters, stable optimal BOP values for $Q_4$, $Q_6$, and $\hat{W}_6$ are reached after about 2600 generations. The $\hat{W}_4$ values (data not shown) do not converge to the optimum for the $CK2_{20}$ cluster. Such variation has also been observed for other instances. We therefore suggest to use the triplet $Q_4$, $Q_6$, and $\hat{W}_6$ as a structural fingerprint for general CK clusters. For efficient restart strategies, it is conceivable to define (local) convergence of an algorithm in structure space in terms of these bond-order parameters rather than in terms of the original variables (i.e., the spherical coordinates).

The failure of the Nelder-Mead algorithm on the $CK2_{20}$ cluster problem suggests that CK clusters are a non-trivial test case for black-box optimizers. For now, we can only speculate about the poor performance of NM on this problem since NM and CMA-ES have the same invariance properties. Hansen tested a restart NM algorithm on the BBOB 2009 test suite and concluded that NM with restarts "allows searching unstructured multi-modal landscapes comparatively effective, while a global topography within a multi-modal or rugged landscape is not well exploited" (Hansen, 2009a). The robust performance of CMA-ES suggests a global single-funnel topology of $CK2_{20}$ with local minima on a smaller scale.

### *Putative ground states of CK1 and CK2 clusters for $N = 2, \ldots, 25$*

So far we have analyzed two CK cluster instances with proven ground states. These instances have dimensionality $n = 16$ and $n = 40$, respectively. In order to construct a benchmark that spans a wider range of dimensions we now consider all CK1 and CK2 clusters for even $n = 4, \ldots, 50$, i.e., clusters containing up to $N = 25$ atoms. Given the promising performance of CMA-ES on the previous instances, we use it as a tool for identifying putative ground states and low-energy local minima. We define the following computational experiments: For each instance, we run 25 standard CMA-ES runs *without* restarts until any of the standard convergence criteria are met. The initial step size is $\sigma_{\text{init}} = 0.4\pi$. The FES budget is restricted to $\text{MAX\_FES} = 10^4 n$. We store all putative global and local optima, energy values, and the number of FES CMA-ES needed to converge. We also calculate the values of the BOPs $Q_4$, $Q_6$, and $\hat{W}_6$ for all observed structures.

We first report the results for CK1 clusters. The energies $E_{\text{CK1}}$ of the putative ground states scale exponentially with cluster size. A least-squares fit results in $E_{\text{CK1}}(\mathbf{x}_{min}) \approx 0.9205 \cdot \exp(0.2328\, N)$ (see Fig. 6.5). The number of FES CMA-ES needed until convergence increases with $N$. The average number of FES scales linearly with cluster size. For $N = 2 \ldots 12, 17, 20, 21, 24$, the variance is very low. For the other instances, however, some CMA-ES runs need considerably more FES to converge than others. This indicates that some problem instances exhibit considerably more complex landscapes than others, and that this phenomenon is not completely determined by the problem dimension. Nevertheless, all runs converge far before exhausting the FES budget. We summarize the information about putative CK1 ground states and low-energy minima as identified by CMA-ES in Table 6.3. We report the energies along with the BOP values for all minima. For $N = 10, 14, 16, 22, 23, 25$, multiple low-energy minima were identified. From the wealth of generated data we discuss three instances in more detail.

The first instance is the $CK1_{12}$ cluster. Its putative ground state is a Mackay icosahedron with 20 triangular faces (see Fig. 6.12) and the well-known BOP pattern $Q_4 = 0$, $Q_6 = 0.0415$, and $\hat{W}_6 = -0.1698$. The 13-atom Lennard-Jones cluster that is discussed in the next section exhibits the same symmetry with a central atom at the origin. From the CMA-ES runtimes we see that $CK1_{12}$ can be found rapidly (in less than $2.5 \cdot 10^4$ FES on average) and robustly (all runs converge to the putative ground state).

| **N** | **Energy** | $Q_4$ | $Q_6$ | $\hat{W}_6$ |
|---|---|---|---|---|
| 2 | 0.1084 | 1 | 1 | -0.0931 |
| 3 | 0.4630 | 0.3750 | 0.7408 | -0.0463 |
| 4 | 1.0583 | 0.5092 | 0.6285 | 0.0132 |
| 4 | 1.1029 | 0.5619 | 0.4369 | 0.0076 |
| 5 | 1.9838 | 0.6250 | 0.4556 | 0.0466 |
| 6 | 3.1501 | 0.7638 | 0.3536 | 0.0132 |
| 7 | 4.6241 | 0.5118 | 0.2861 | 0.0598 |
| 8 | 6.3434 | 0.5092 | 0.6285 | 0.0132 |
| 9 | 8.3580 | 0.1387 | 0.3561 | -0.0342 |
| 10A | 10.6645 | 0.2574 | 0.3289 | 0.0407 |
| 10B | 10.6646 | 0.2740 | 0.3651 | 0.0254 |
| 11 | 13.3828 | 0.0198 | 0.1129 | 0.1293 |
| 12 | 16.1847 | 0 | 0.0415 | -0.1698 |
| 13 | 20.5410 | 0.0736 | 0.2470 | -0.0032 |
| 14A | 25.1733 | 0.0771 | 0.2328 | 0.0131 |
| 14B | 25.1880 | 0.0729 | 0.2212 | 0.0111 |
| 14C | 25.2088 | 0.0278 | 0.0285 | -0.0931 |
| 15 | 30.9350 | 0.0332 | 0.1080 | 0.1196 |
| 16A | 38.4985 | 0.0464 | 0.0090 | 0.0931 |
| 16B | 38.5386 | 0.0615 | 0.2061 | -0.0631 |
| 17 | 47.3064 | 0.1044 | 0.0509 | 0.0931 |
| 18 | 59.9795 | 0.0052 | 0.1623 | -0.0931 |
| 19 | 78.2895 | 0.0487 | 0.1009 | 0.1476 |
| 20 | 94.1138 | 0.0630 | 0.1052 | -0.0407 |
| 21 | 122.3120 | 0.0173 | 0.1709 | -0.0274 |
| 22A | 151.7772 | 0.0294 | 0.0043 | 0.0132 |
| 22B | 153.1696 | 0.0307 | 0.0481 | 0.0026 |
| 23A | 202.9820 | 0.0139 | 0.1418 | -0.0329 |
| 23B | 203.0328 | 0.0145 | 0.1410 | -0.0277 |
| 24 | 236.1115 | 0.0164 | 0.0363 | 0.0132 |
| 25A | 314.0809 | 0.0047 | 0.1300 | -0.0158 |
| 25B | 314.0909 | 0.0048 | 0.1299 | -0.0176 |
| 25C | 321.7856 | 0.0235 | 0.1466 | -0.0316 |

Table 6.3: Summary statistics of the putative CK1 ground states and low-energy minima as identified by CMA-ES. The number of particles, energy, and the BOP values $Q_4$, $Q_6$, and $\hat{W}_6$ are reported for all instances. For $N = 10, 14, 16, 22, 23, 25$, multiple minima were identified. The putative ground states are labeled "A", other minima "B" or "C".

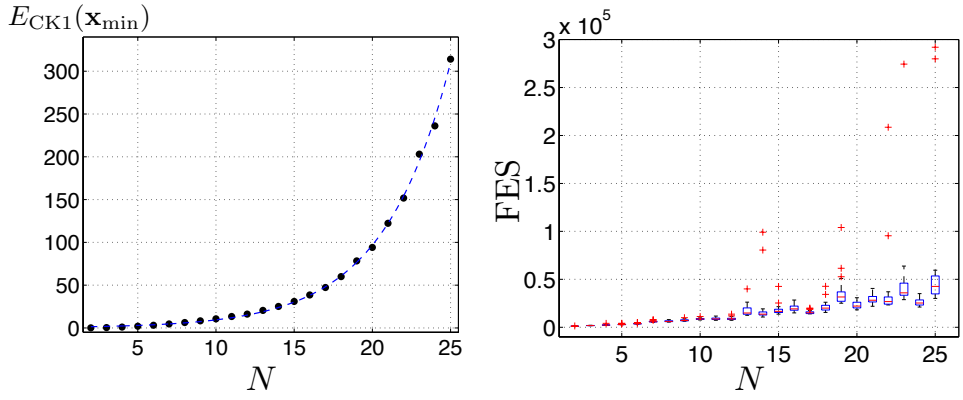



Figure 6.5: The left panel shows $E_{\mathrm{CK1}}$ of the putative ground states versus the number of atoms in the cluster $N$. The dots represent the results from the CMA-ES runs, the dashed curve is the best exponential least-squares fit. The right panel shows box plots of the numbers of FES needed until CMA-ES converges for the different $N$.

For $\mathrm{CK1}_{14}$ clusters, CMA-ES converges to three different minima. In 21 out of the 25 runs CMA-ES identifies the putative ground state (labeled 14A) with energy 25.1733. Two runs converge to a low-energy local minimum (labeled 14B) with energy 25.1880 and two runs to a local minimum (labeled 14C) with energy 25.2088. The corresponding configurations are shown in Fig. 6.6. We highlight this cluster instance because the putative ground state might

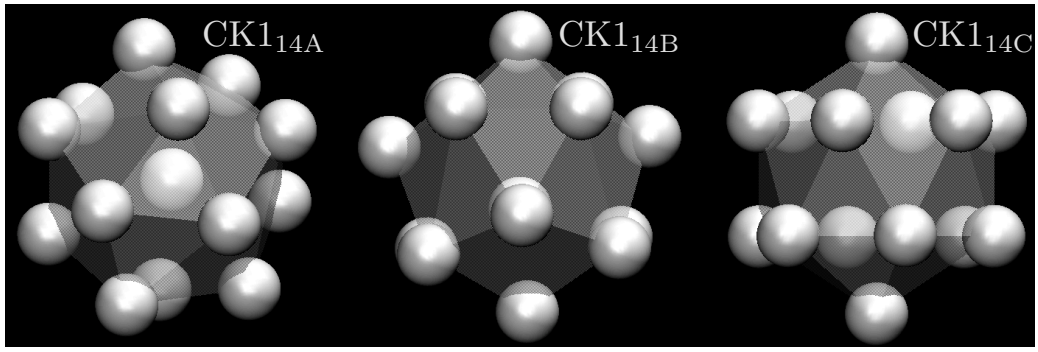



Figure 6.6: Putative ground state configuration ($\mathrm{CK1}_{14\mathrm{A}}$) and two low-energy stable configurations ($\mathrm{CK1}_{14\mathrm{B}}$ and $\mathrm{CK1}_{14\mathrm{C}}$) of the $\mathrm{CK1}_{14}$ cluster.

seem counterintuitive at first. A human observer would possibly favor structures 14B and 14C over 14A due to their apparent symmetry. They are, however, higher in energy than the putative ground state 14A. Moreover, structure 14A attains a value of $\hat{W}_6 = 0.0131$, indicative of maximum cubic symmetry (Steinhardt et al., 1983).

The energy landscape of $\mathrm{CK1}_{16}$ clusters exhibits two competing low-energy structures as depicted in Fig. 6.7. 18 out of the 25 CMA-ES runs converge to structure 16A, the remaining 7 runs find structure 16B. $\mathrm{CK1}_{16\mathrm{A}}$ consists of two opposite, rotated square faces (like

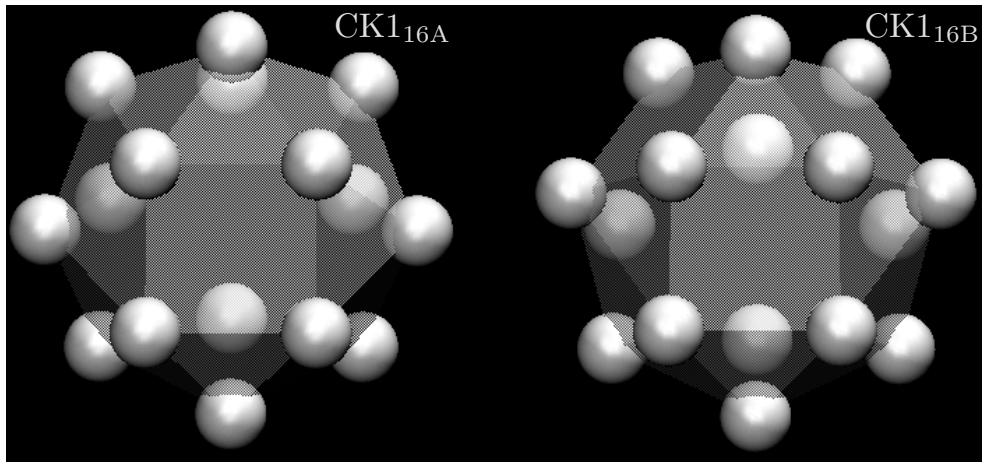



Figure 6.7: Putative ground state configuration ($CK1_{16A}$) and a competing sub-optimal configuration ($CK1_{16B}$) of the $CK1_{16}$ cluster.

in the anti-cube), and triangular faces otherwise. $CK1_{16B}$ has three square faces grouped around a central triangle, and otherwise triangular ones, leading to a different set of BOP values (see Table 6.3). In order to check whether the energy landscape explored by CMA-ES exhibits a single-funnel topology with two low-energy minima at the bottom, or rather a double-funnel landscape, we conduct the following experiment: We start CMA-ES runs from the sub-optimal low-energy structure as initial configuration with increasing initial $\sigma$ values. These experiments reveal how much the initial configuration needs to be perturbed until CMA-ES is able to detect the putative globally optimal solution. We choose $\sigma_{\text{init}} \in \{0.001\pi, 0.01\pi, 0.025\pi, 0.05\pi, 0.075\pi, 0.1\pi, 0.125\pi, 0.175\pi, 0.2\pi\}$ and repeat the experiment 50 times per $\sigma_{\text{init}}$. We monitor whether CMA-ES returns to the sub-optimal solution or enters the putative ground state. The frequency of transition serves as an estimator for the transition probability $\mathrm{P}(CK1_{16B} \rightarrow CK1_{16A})$ under CMA-ES exploration, and hence for the relative basin size of the sub-optimal structure. The $\sigma$-dependent transition probability and a typical trajectory of CMA-ES in $Q_6 - \hat{W}_6$ space leading from the sub-optimal basin to the putative optimal basin are shown in Fig. 6.8. The experiment suggests that below $\sigma_{\text{init}} = 0.01\pi$, CMA-ES does not leave the basin of the sub-optimal solution. For larger $\sigma_{\text{init}}$ the probability increases until it reaches a similar level as with global CMA-ES settings. Using $\sigma_{\text{init}} = 0.07\pi$, the probability is about 1/2 to fall into either minimum. The example CMA-ES

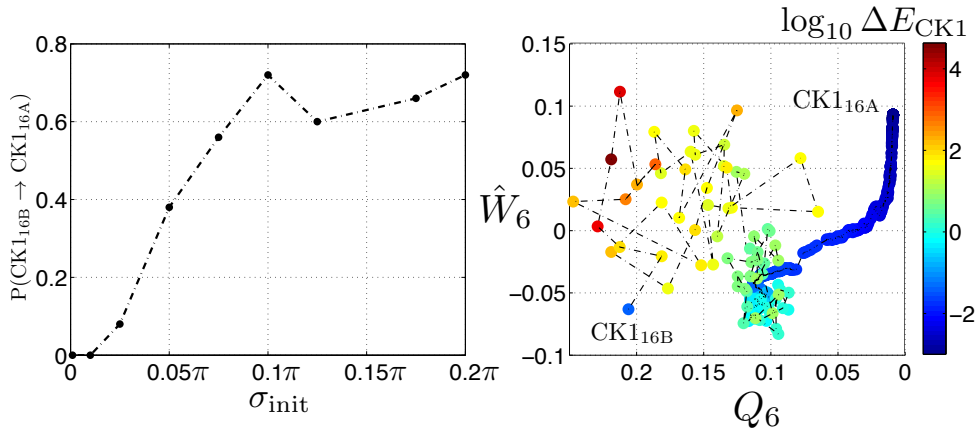



Figure 6.8: The left panel shows the $\sigma$-dependent transition probability P(16B $\to$ 16A) for CMA-ES. The right panel depicts a typical trajectory of CMA-ES' mean in the $Q_6 - \hat{W}_6$ plane for $\sigma_{\text{init}} = 0.1\pi$. Each configuration is color-coded by the $\log_{10} \Delta E_{\text{CK1}} = \log_{10}(E_{\text{CK1}}(\mathbf{x}) - E_{\text{CK1}}(\mathbf{x}_{\text{min}}))$.

trajectory shown in the right panel of Fig. 6.8 for $\sigma_{\text{init}} = 0.1\pi$ reveals an interesting pattern in $Q_6 - \hat{W}_6$ space. Starting from the 16B structure, CMA-ES first performs a random walk until it clusters around configurations with BOP values $Q_6 \approx 0.12$, $\hat{W}_6 \approx -0.05$, which most probably include the transition state between the two minima. The trajectory then smoothly converges to the final BOP values of the 16A structure. Trajectories that return to the sub-optimal structure 16B behave similarly, with a cluster at $Q_6 \approx 0.15, \hat{W}_6 \approx -0.05$ before smoothly converging back to the BOP values of the 16B configuration (data not shown). In summary, our experiments suggest that the $CK1_{16}$ cluster landscape under CMA-ES exhibits a single-funnel topology with two competing minima at the bottom of the funnel. The putative optimal configuration is located in a considerably larger basin, hence representing a moderately difficult problem for CMA-ES. Nevertheless, it shall be interesting to test other search heuristics on this problem, especially with respect to their susceptibility to the competing sub-optimal solution.

We now present some results for CK2 clusters. The scaling of the minimum energy with cluster size is shown in Fig. 6.9. For the cluster sizes considered, the energies $E_{\text{CK2}}$ of the putative ground states scale quadratically with cluster size. The best least squares fit is achieved by $E_{\text{CK2}}(\mathbf{x}_{min}) \approx 2.764N^2 - 19.58\,N + 31.23$. This surprising result can be explained by the moderate increase of the pair potential in the range of the observed distances for these cluster sizes (see Fig. 6.1). Addition of a single particle hence only results in a quadratic increase of the energy. The average number of FES required by CMA-ES to converge to the putative minimum scales linearly with cluster size up to $N = 12$. For $N = 2, \ldots, 10, 12, 14, 17$, the variance is very low. Compared to CK1 clusters, the number of FES needed to converge is considerably higher for larger CK2 clusters (well above $5 \cdot 10^4$ FES on average), but is still an order of magnitude below the allowed FES budget. We summarize the information about putative

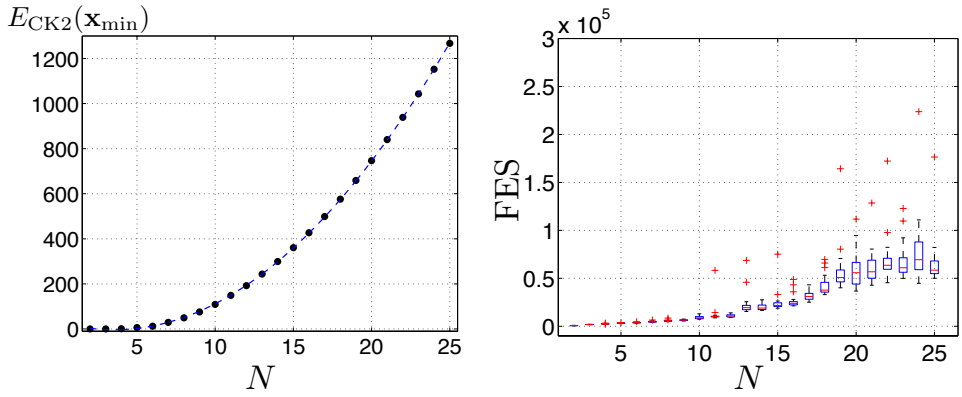



Figure 6.9: The left panel shows $E_{\mathrm{CK2}}$ of the putative ground states versus the number of atoms in the cluster $N$. The dots represent the results from the CMA-ES runs, the dashed curve is the best quadratic least squares fit. The right panel shows box plots of the number of FES needed until CMA-ES converges for the different $N$.

ground states and local minima of CK2 clusters as identified by CMA-ES in Table 6.4. The energies and BOP values of all detected minima are reported. For $N = 4, 13, 16, 19, 21, \ldots, 25$, multiple minima were found. While these could be discussed analogously to our findings for CK1 clusters, we restrict ourselves to the interesting case of 4-atom CK2 clusters. In 24 out of the 25 runs CMA-ES finds the putative optimal ground state 4A, which is a regular tetrahedron with all pairwise distances equal to $\sqrt{8/3}$. In one run, CMA-ES finds a high-energy local minimum, where the atoms form a pyramid with a larger triangular base face and three smaller triangular side faces (4B). The atoms form three distances of length $\sqrt{3}$ and three distances of length $\sqrt{2}$. Both structures are depicted in Fig. 6.10. It is surprising that the

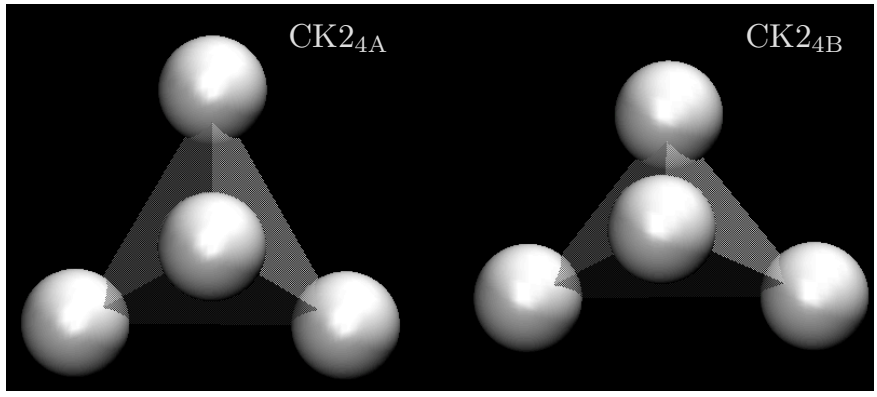



Figure 6.10: Putative tetrahedral ground state configuration ($\mathrm{CK1}_{4\mathrm{A}}$) and a competing sub-optimal pyramidal configuration ($\mathrm{CK2}_{4\mathrm{B}}$) of the $\mathrm{CK2}_4$ cluster.

strictly convex CK2 pair potential produces a non-convex energy landscape even in the 4-atom

| **N** | **Energy** | $Q_4$ | $Q_6$ | $\hat{W}_6$ |
|---|---|---|---|---|
| 2 | 0 | 1 | 1 | -0.0931 |
| 3 | 0.0939 | 0.3750 | 0.7408 | -0.0463 |
| 4A | 0.7901 | 0.5092 | 0.6285 | 0.0132 |
| 4B | 3.0958 | 0.5312 | 0.5040 | 0.0048 |
| 5 | 6.0977 | 0.6250 | 0.4556 | 0.0466 |
| 6 | 12.0076 | 0.7638 | 0.3536 | 0.0132 |
| 7 | 29.2253 | 0.5536 | 0.0625 | -0.0931 |
| 8 | 49.3528 | 0.3736 | 0.2502 | -0.0931 |
| 9 | 75.8984 | 0.1502 | 0.3391 | -0.0436 |
| 10 | 108.7305 | 0.1702 | 0.1579 | -0.0931 |
| 11 | 149.0286 | 0.1043 | 0.3329 | -0.0515 |
| 12 | 192.0350 | 0 | 0.0415 | -0.1698 |
| 13A | 243.5499 | 0.0780 | 0.2679 | 0.0066 |
| 13B | 243.5531 | 0.1199 | 0.2714 | 0.0115 |
| 14 | 298.9590 | 0.2813 | 0.5036 | 0.0132 |
| 15 | 360.5035 | 0.0875 | 0.2977 | 0.0076 |
| 16A | 426.8723 | 0.0225 | 0.2791 | -0.0219 |
| 16B | 426.8726 | 0.0352 | 0.2890 | -0.0258 |
| 17 | 499.0473 | 0.1055 | 0.1994 | 0.0019 |
| 18 | 576.1469 | 0.1608 | 0.3495 | 0.0566 |
| 19A | 658.8684 | 0.1027 | 0.2900 | 0.1166 |
| 19B | 658.8689 | 0.0929 | 0.2822 | 0.1205 |
| 20 | 746.6667 | 0 | 0.2718 | 0.1698 |
| 21A | 840.1743 | 0.0389 | 0.1921 | 0.1141 |
| 21B | 840.1976 | 0.0372 | 0.1824 | 0.0835 |
| 21C | 840.2036 | 0.0364 | 0.1575 | 0.0922 |
| 22A | 938.8178 | 0.0353 | 0.1494 | 0.1517 |
| 22B | 938.8197 | 0.0326 | 0.1782 | 0.1098 |
| 23A | 1042.8819 | 0.0335 | 0.0571 | -0.0206 |
| 23B | 1042.8846 | 0.0193 | 0.0592 | -0.0507 |
| 23C | 1042.8900 | 0.0185 | 0.1077 | -0.0846 |
| 23D | 1042.9105 | 0.0181 | 0.1098 | 0.0301 |
| 24A | 1152.1594 | 0.0058 | 0.0153 | 0.0132 |
| 24B | 1152.1789 | 0.0037 | 0.0050 | -0.0931 |
| 25A | 1266.8947 | 0.0130 | 0.0852 | 0.0706 |
| 25B | 1266.9774 | 0.0186 | 0.0201 | 0.0931 |

Table 6.4: Summary statistics of the putative CK2 ground states and local minima as identified by CMA-ES. The number of particles, energy, and the BOP values $Q_4$, $Q_6$, and $\hat{W}_6$ are reported for all instances. For $N = 4, 13, 16, 19, 21, \ldots, 25$, multiple minima are identified. The putative ground states are labeled "A".

case. Although the 4B structure is much higher in energy (3.0958) than the 4A tetrahedron (0.7901), its basin is relatively stable. Similar transition path experiments as were done for the $CK1_{16}$ cluster reveal that when starting CMA-ES from the pyramidal structure, there is a high probability to converge back to the sub-optimal structure even for $\sigma_{\text{init}}$ as high as $0.4\pi$. Nevertheless, given the empirical hitting probability of $1/25$, the overall basin size is negligible for CMA-ES.

In summary, we presented a detailed analysis of Cohn-Kumar clusters arising from two different strictly convex pair potentials. We analyzed the configurations where proven global minima exist and extended to other instances for up to $N = 25$ atoms. In order to show the richness of the energy landscapes we analyzed several instances in further detail using CMA-ES as a search heuristic and Steinhardt's bond-order parameters as measures to characterize the found energy minima. Table 6.5 summarizes our suggested CK cluster test suite settings.

| Problems | CK1, CK2 |
|---|---|
| Runs per problem | 25 |
| $n$ | $4, 6, 8, \ldots, 50$ |
| MAX_FES | $10000 \cdot n$ |
| Termination | If FES = MAX_FES or $E_{\text{CKi}}(\mathbf{x}) \leq E_{\text{CKi}}(\mathbf{x}_{\min}) + 10^{-6}$ |
| Initialization and bounds | Uniformly random in $[-\pi, \pi]^n$ |

Table 6.5: Suggested benchmark settings for the CK1/CK2 cluster test suite.

## 6.4 Lennard-Jones clusters

Energy landscapes of collections of atoms that interact according to the Lennard-Jones (LJ) pair potential are among the best studied models in theoretical cluster chemistry and biophysics. In cluster chemistry, the LJ potential is widely used to model the behavior of noble gases such as Argon. Biophysicists use the LJ potential to model the hydrophobic forces in biopolymers such as proteins and alkanes. The problem of finding minimum-energy configurations of LJ clusters has fascinated researchers for over three decades and is regularly used as a standard test case for first-order search heuristics. In LJ clusters, each pair of atoms interacts through the following pair potential:

$$u_{\text{LJ}}(r_{ij}) = 4\epsilon \left( \left( \frac{\sigma_{\text{LJ}}}{r_{ij}} \right)^{12} - \left( \frac{\sigma_{\text{LJ}}}{r_{ij}} \right)^{6} \right), \tag{6.11}$$

where $r_{ij} = d_{\text{E}}(\mathbf{p}_i, \mathbf{p}_j)$ and $\mathbf{p}_i = (x_i, y_i, z_i)$ the 3D position of atom $i$. The parameter $\epsilon$ is the potential well depth (in units of energy) and $2^{\frac{1}{6}}\sigma_{\text{LJ}}$ is the equilibrium inter-atom distance (in units of length) at zero temperature. Figure 6.11 depicts the unimodal shape of the LJ pair

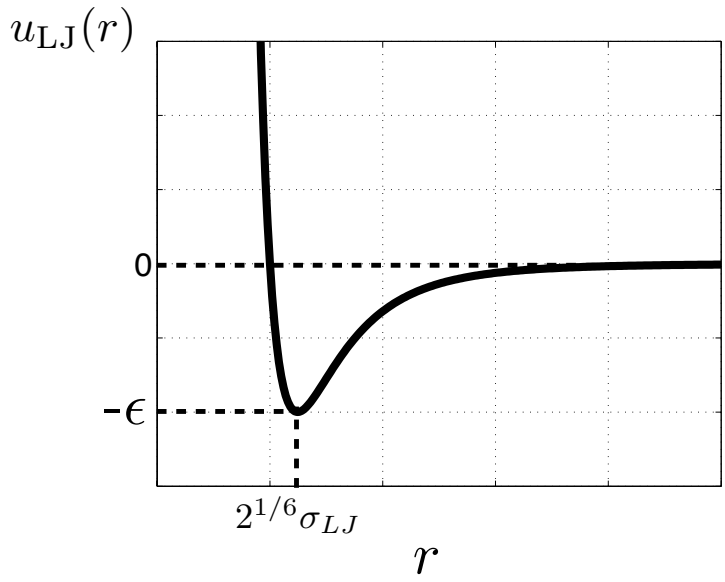



Figure 6.11: The Lennard-Jones pair potential $u_{\mathrm{LJ}}(r)$ versus distance $r$. The minimum is at $r = 2^{1/6}\sigma_{\mathrm{LJ}}$ with energy $-\epsilon$. For $r \to \infty$, the potential asymptotically approaches 0.

potential and the role of the parameters. The potential energy $E_{\mathrm{LJ}}$ of a cluster of $N$ LJ atoms is given by:

$$E_{\mathrm{LJ}}(\mathbf{x}) = 4\epsilon \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \left( \left( \frac{\sigma_{\mathrm{LJ}}}{r_{ij}} \right)^{12} - \left( \frac{\sigma_{\mathrm{LJ}}}{r_{ij}} \right)^{6} \right) . \tag{6.12}$$

Again, $\mathbf{x} = \{\mathbf{p}_1, \ldots, \mathbf{p}_i, \ldots, \mathbf{p}_N\}$ and $r_{ij} = d_{\mathrm{E}}(\mathbf{p}_i, \mathbf{p}_j)$. The ambient space of the atoms is the 3D Euclidian space. Knowledge about minimum-energy (or ground-state) configurations of LJ clusters allows predicting properties of crystallization or solid–liquid transitions of noble atomic mixtures at low temperatures. The presence of a well in the LJ pair potential implies that a collection of atoms faces the problem of frustration. Although all atoms "like" to have their neighbors at equilibrium distance, there is no geometric configuration that could achieve this for $N > 4$. This introduces multi-modality in the landscape with competing low-energy configurations.

For a long time it was believed that ground states of LJ clusters could be efficiently constructed by *aufbau* algorithms (Hoare, 1979). Starting from a "seed" structure with specific symmetry and $N - 1$ atoms, these algorithms construct a putative ground state of $N$ atoms by placing an additional atom at the energetically most favorable location and relaxing the resulting structure by energy-gradient descent. In the past 25 years, however, it has been shown that such algorithms are not able to identify many of today's known putative ground states. Wille identified the putative ground state of the $\mathrm{LJ}_{24}$ cluster using Simulated Annealing with a specialized problem-specific move set (Wille, 1987). This is the only ground state so far that has first been found by a method that does not use of gradient information. Northby identified

putative ground states for $13 \leq N \leq 147$ by searching lattices with icosahedral symmetry (Northby, 1987) and gradient minimization. However, not all LJ instances follow an icosahedral symmetry, a fact that has been discovered only in the mid-1990's. The ground states of LJ clusters with $N = 38, 75 - 77, 98, 101 - 103$ atoms follow different packing schemes. They have mostly been identified by extensive application of "unbiased" gradient-based optimizers such as the hybrid genetic algorithm by Deaven and co-workers (Deaven et al., 1996) or Basin-Hopping (Li and Scheraga, 1987; Wales and Doye, 1997). The late discovery of these ground states is explained by the deceiving landscape topology with the global minimum located in a narrow funnel. In Section 6.4.1 we consider the "archetypal" (Wales and Scheraga, 1999) double-funnel energy landscape of the $LJ_{38}$ cluster and present a tuning technique that is able to smoothly deform the topology to a single-funnel problem.

While large LJ clusters are prohibitive for black-box optimization benchmarking due to their staggering number of local minima, we nonetheless propose small instances with up to $N \leq 19$ atoms as meaningful benchmark problems. Minimizing the potential energy of a cluster of $N$ atoms in 3D space defines a continuous optimization problem in $n = 3N - 6$ dimensions, since 3 translational and 3 rotational degrees of freedom can be removed from the system. This is achieved by placing the first atom at the origin of the Cartesian coordinate system, the second along the $x$-axis, and the third in the $xy$-plane. Hence, $N = 19$ defines a problem in $n = 51$ dimensions. We characterize the putative ground states of LJ clusters with $N \leq 19$ in the following subsection and present numerical optimization runs for selected instances. We emphasize that LJ clusters have, despite their widespread use in chemistry and physics, not been subject to rigorous studies using black-box heuristics. We are only aware of two publications where certain small cluster instances have been optimized using evolutionary algorithms: Müller and co-workers presented some initial results for CMA-ES on LJ clusters with $N = 8, 27$ (Müller et al., 2003), and Call and co-workers optimized $LJ_{26}$ with a specialized PSO (Call et al., 2007).

### *Lennard-Jones clusters for $N \leq 19$*

All known putative ground states of LJ clusters for the standard parametrization $\epsilon = \sigma_{\mathrm{LJ}} = 1$ are available from the Cambridge Cluster Database (Wales et al., 2009). We characterize the structures using the BOP parameters $Q_4$, $Q_6$, and $\hat{W}_6$ with the suggested $r_0 = 1.391$ for LJ clusters (Doye et al., 1999b). The data are summarized in Table 6.6. Most of the structures have a low $Q_4$ value, indicating icosahedral symmetry (Steinhardt et al., 1983). As previously mentioned, the $LJ_{13}$ ground state is identical (with identical BOP values) to the putative $CK1_{12}$ ground state, but with an additional central atom at the origin (see also Fig. 6.12). This instance is also the first of the so-called "magic number" structures. Magic number LJ clusters are based on *complete* multilayer Mackay icosahedra with $N = (2i + 1)(5/3\, i(i + 1) + 1), i = 1, 2, 3, \ldots$ atoms. They are so stable that they are regularly found in NMR experiments, for instance of Xeon clusters (Hasse, 1991). We provide here numerical experiments on three select instances: $LJ_7$, $LJ_{13}$, and $LJ_{19}$. The putative ground state structures of these clusters are depicted in Fig. 6.12. We focus on these three instances since they have been extensively analyzed in (Wales, 2005). $LJ_{13}$ and $LJ_{19}$ are also considered in (Doye et al., 1999a). Detailed information is available about the number of unique local minima, first- and higher-order saddle

| **N** | **Energy** | $Q_4$ | $Q_6$ | $\hat{W}_6$ |
|---|---|---|---|---|
| 3 | -3 | 0.3750 | 0.7408 | -0.0463 |
| 4 | -6 | 0.1909 | 0.5745 | -0.0132 |
| 5 | -9.1039 | 0.0013 | 0.4297 | 0.0314 |
| 6 | -12.7121 | 0.1909 | 0.5745 | -0.0132 |
| 7 | -16.5054 | 0.0148 | 0.1604 | -0.0931 |
| 8 | -19.8215 | 0.0644 | 0.1467 | 0.0015 |
| 9 | -24.1134 | 0.0156 | 0.1226 | -0.0391 |
| 10 | -28.4225 | 0.0382 | 0.1359 | -0.0349 |
| 11 | -32.7660 | 0.0298 | 0.1384 | 0.0485 |
| 12 | -37.9676 | 0.0178 | 0.1186 | 0.1209 |
| 13 | -44.3268 | 0 | 0.0415 | -0.1698 |
| 14 | -47.8452 | 0.0283 | 0.0437 | 0.0128 |
| 15 | -52.3226 | 0.0069 | 0.0437 | -0.0790 |
| 16 | -56.8157 | 0.0208 | 0.0653 | -0.0888 |
| 17 | -61.3180 | 0.0216 | 0.0849 | -0.0778 |
| 18 | -66.5309 | 0.0148 | 0.0676 | 0.1613 |
| 19 | -72.6598 | 0.0043 | 0.0056 | 0.0931 |

Table 6.6: Summary statistics of the putative LJ ground states from the Cambridge Cluster Database (Wales et al., 2009). The number of particles, energy, and the BOPs $Q_4$, $Q_6$, and $\hat{W}_6$ are given for each instance.

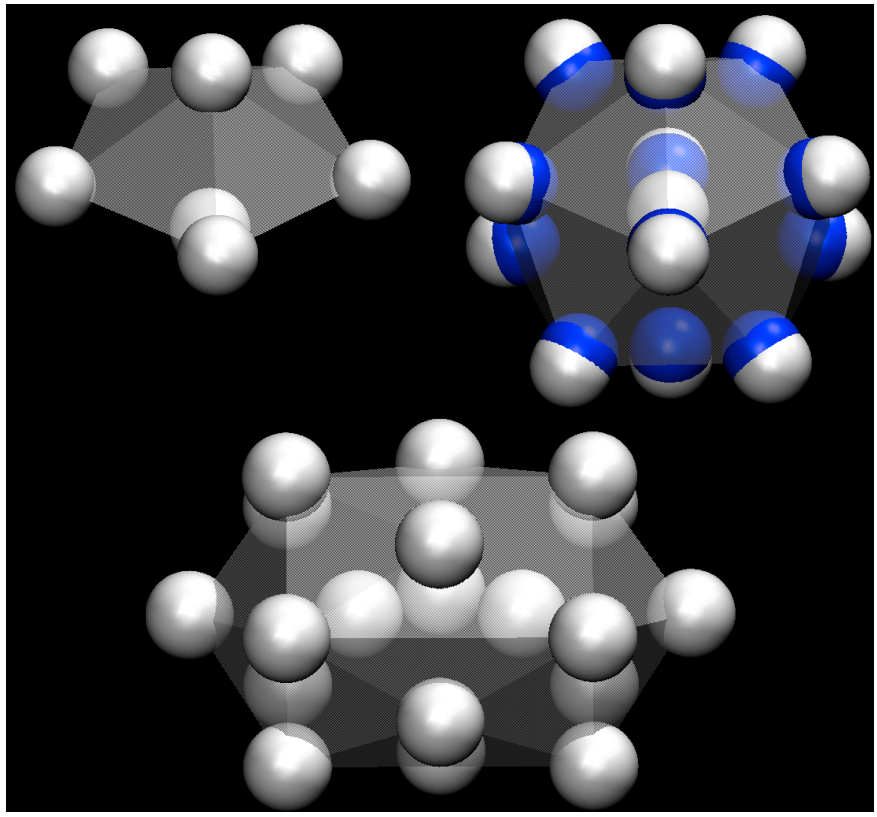


Figure 6.12: Putative ground states for LJ clusters with 7 atoms (top left), 13 atoms (top right), and 19 atoms (bottom). The packing of $LJ_{13}$ is a Mackay icosahedron. The putative optimal $CK1_{12}$ cluster, overlaid in blue, shares the same structure.

points, as well as disconnectivity graphs of low-energy minima. Prior analysis also showed that the energy landscapes of these clusters exhibit single-funnel topologies, which should render them feasible for many black-box optimizers, despite the huge number of local minima. We imagine these instances as cluster analogs of Rastrigin-like test functions. For the numerical experiments we suggest the benchmark settings summarized in Table 6.8. We run IPOP-CMA-ES with standard settings, yet with a smaller increase factor for the population size ($incPop = 1.25$). The statistics about the number of FES and the success rate $p_s$ for CMA-ES runs on $LJ_7$, $LJ_{13}$, and $LJ_{19}$ are summarized in Table 6.7. CMA-ES always finds the optimal

| | **min** | **7th** | **median** | **19th** | **max** | **mean** | **std** | $p_s$ |
|---|---|---|---|---|---|---|---|---|
| $\mathbf{LJ}_7$ | 3805 | 5509 | 15108 | 23980 | 89819 | 18220.36 | 18218.27 | 1 |
| $\mathbf{LJ}_{13}$ | 13007 | 51827 | 101726 | 178359 | - | 109377.12 | 80713.15 | 0.96 |
| $\mathbf{LJ}_{19}$ | 31726 | - | - | - | - | 270870.50 | 338201.40 | 0.08 |

Table 6.7: Statistics of the number of FES for CMA-ES runs that reach the globally optimal configurations for the $LJ_7$, $LJ_{13}$, and $LJ_{19}$ cluster problems, along with their success rates $p_s$.

structure of $LJ_7$. In 24 out of the 25 runs, it also solves the $LJ_{13}$ cluster problem. Inspection of Doye's and Wales's disconnectivity graphs (see Fig. 6.13 left panel) for $LJ_{13}$ reveals that CMA-ES converges to the second-best optimum in the one case where it does not find the global optimum. The $LJ_{19}$ problem is more challenging. Only in 2 out of the 25 runs CMA-ES is able to identify the putative ground state. In the other runs, CMA-ES detects twice the second-best minimum and in the remaining cases a diverse set of local low-energy minima (see Fig. 6.13 right panel). These results confirm that LJ clusters with moderate numbers of atoms are amenable to black-box optimization, but are considerably more difficult for CMA-ES than CK1/CK2 clusters with the same number of degrees of freedom. In benchmark scenarios with a limited function evaluation budget, LJ clusters with larger numbers of atoms are likely to be hard for black-box search.

| Problems | LJ |
|---|---|
| Runs per problem | 25 |
| $n$ | $3, 6, 9, \ldots, 51$ |
| MAX_FES | $10000 \cdot n$ |
| Termination | If FES = MAX_FES or<br>$E_{\mathrm{LJ}}(\mathbf{x}) \leq E_{\mathrm{LJ}}(\mathbf{x}_{\min}) + 10^{-6}$ |
| Initialization and bounds | Uniformly random in $[-2, 2]^n$ |

Table 6.8: Suggested benchmark settings for the LJ cluster test suite.

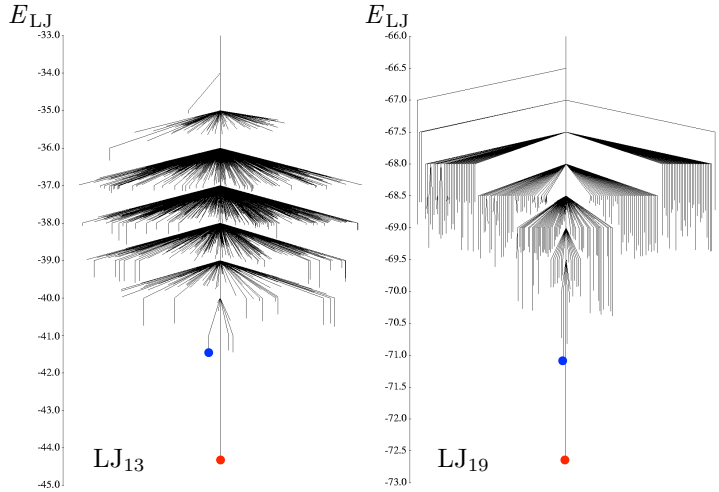



Figure 6.13: Disconnectivity graph of all minima of $\mathrm{LJ}_{13}$ and of the 250 lowest-lying minima of $\mathrm{LJ}_{19}$ found by the BH algorithm (adapted from (Doye et al., 1999a)). The red dots indicate the ground states, the blue dots the second-best minima. These ground states and minima were also found by CMA-ES.

### 6.4.1 The $\mathrm{LJ}_{38}$ cluster as a high-dimensional benchmark with tunable landscape topology

All cluster instances introduced so far have a static landscape topology that is solely determined by the underlying pair potential and the number of atoms. We complement the set of cluster benchmarks with a problem instance that (i) is high-dimensional, but solvable in reasonable time, and (ii) exhibits a tunable landscape topology. The benchmark is based on the $\mathrm{LJ}_{38}$ cluster problem, which has also been the inspiration for Lunacek's double-funnel test case (Whitley, 2010). We provide a complete description of the proposed benchmark. An initial description with extensive numerical optimization runs using IPOP-CMA-ES has been reported in (Müller and Sbalzarini, 2009). The topology of the standard $\mathrm{LJ}_{38}$ energy landscape has been widely studied in the literature (Barron et al., 1996; Doye et al., 1999b; Wales, 2004). It exhibits a double-funnel structure where the global minimum-energy configuration with face-centered cubic octahedral (fcc) symmetry lies in the narrow funnel. The majority of local minima, most of them with icosahedral (ico) symmetry, populate the wider funnel. Figure 6.14a shows a sketch of this landscape. Leary (Leary, 2000) estimated the size of the optimal funnel from Monotonic Sequence BH runs to be around 12.4% of the entire configuration space. For 1000 randomly generated configurations, he applied the BH move set and only accepted improving configurations. In 124 out of the 1000 runs BH reached the fcc structure. For standard BH, however, the probability is much lower since most runs converge to structures within the larger, sub-optimal funnel. In fact, the optimal fcc structure has originally not been

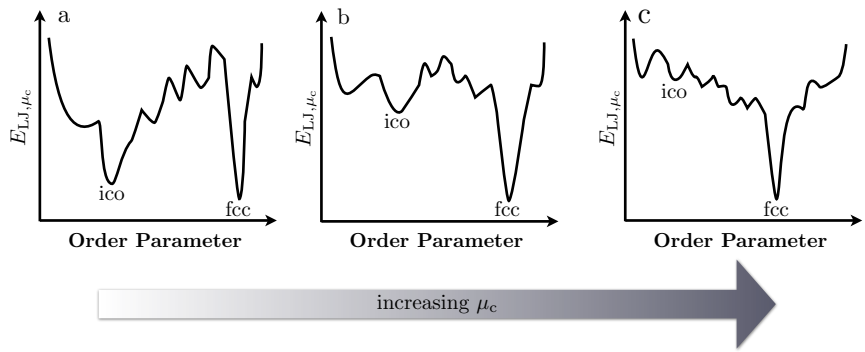



Figure 6.14: Sketch of the $\mu_c$-dependent evolution of the $LJ_{38}$ energy landscape. The $x$-axis represents a suitable order parameter that can discriminate between different cluster topologies, the $y$-axis represents the potential energy $E_{LJ,\mu_c}$. The global topology of the landscape is a double funnel with a sub-optimal ico structure at the bottom of the wider funnel. The narrow funnel contains the global minimum with fcc symmetry. Increasing $\mu_c$ gradually changes the landscape topology from a double-funnel to a single-funnel.

found by unbiased optimization methods, but by a combination of "geometric intuition" and local minimization (Barron et al., 1996). Barron and co-workers first constructed initial $LJ_{38}$ configurations on fcc lattices and then applied gradient-based minimization in order to relax these structures under the LJ pair potential (Barron et al., 1996). Doye and co-workers (Doye et al., 1999b) eventually revealed the "paradigmatic" double-funnel nature of the landscape. They characterized it in terms of the number and location of minima, structural diversity of the minima, and the energy barrier between the two funnels. Based on this information, Doye realized that the problem of finding the global minimum of $LJ_{38}$ can be simplified by introducing a penalty term that simulates "compression" in the original energy function (Doye, 2000). Compression can be seen as a transformation of the PEL that favors more compact structures. This leaves the funnel that contains the more compact fcc structures unchanged and lifts the energies of ico structures until the corresponding funnel vanishes. Doye proposed the following penalized energy function:

$$E_{\mathrm{LJ},\mu_c}(\mathbf{x}) = E_{\mathrm{LJ}}(\mathbf{x}) + \mu_c\, Q_c(\mathbf{x})\,, \tag{6.13}$$

where $Q_c(\mathbf{x}) = \sum_{i=1}^{N} \frac{d_E(\mathbf{p}_i,\mathbf{p}_{cm})^2}{\sigma_{LJ}^2}$ with $\mathbf{p}_{cm}$ the center of mass of the cluster. The compression term has the form of an atomic positional variance. For the best icosahedral structure it is $Q_c(\mathbf{x}^{ico}) = 96.1624$ and for the best fcc structure $Q_c(\mathbf{x}^{fcc}) = 91.6369$, hence distorting the energy difference between the two competing structures in favor of the octahedron. The scalar parameter $\mu_c$ controls the magnitude of compression. The effect of the $\mu_c$-dependent compression on the topology of the PEL is visualized with disconnectivity graphs in (Doye, 2000) and (Wales, 2004), pp. 338–339. When $\mu_c = 0\epsilon$, we recover the original $LJ_{38}$ cluster problem. For $\mu_c \geq 5\epsilon$, the PEL exhibits a clear single-funnel topology. We sketch this phenomenon in Fig. 6.14. Although the compression term also lowers the average barrier between local minima (Doye, 2000), the system still contains a staggering number of local minima for all $\mu_c$ values considered.

We propose the 38-atom LJ cluster with the energy function defined in Eq. (6.13) as a high-dimensional tunable test case to study the performance of black-box optimization methods as a function of landscape topology. In particular, we suggest the following benchmark scenario: First, the budget of allowed FES should be considerably increased. Second, an algorithm should be tested with $\mu_c \in \{0\epsilon, 0.5\epsilon, \ldots, 5\epsilon\}$. The lower the $\mu_c$ for which the algorithm can still find the putative ground state, the less sensitive it is to the landscape topology. We obtain the putative ground states for the different $\mu_c$ using reference CMA-ES runs. Therefore, we start CMA-ES from the known optimal $\mathbf{x}^{\text{fcc}}$ structure of the original problem with a small $\sigma_{init} = 0.001$. From there, it quickly converges to slightly different fcc structures for varying $\mu_c$. The energies and BOP values of the putative ground states are summarized in Table 6.9. For $\mu_c = 1\epsilon, 5\epsilon$ the resulting energies match the ones reported in (Doye, 2000), providing further confidence that the ground states found here are correct. Visual inspection and the

| $\mu_c$ | **Energy** | $Q_4$ | $Q_6$ | $\hat{W}_6$ |
|---|---|---|---|---|
| 0 | -173.92843 | 0.19090 | 0.57446 | -0.01316 |
| 0.5 | -128.42914 | 0.19090 | 0.57445 | -0.01316 |
| 1 | -83.50595 | 0.19090 | 0.57444 | -0.01316 |
| 1.5 | -39.08437 | 0.19091 | 0.57443 | -0.01316 |
| 2 | 4.89246 | 0.19091 | 0.57442 | -0.01316 |
| 2.5 | 48.46946 | 0.19091 | 0.57441 | -0.01316 |
| 3 | 91.68310 | 0.19092 | 0.57440 | -0.01316 |
| 3.5 | 134.56362 | 0.19092 | 0.57439 | -0.01316 |
| 4 | 177.13651 | 0.19092 | 0.57438 | -0.01316 |
| 4.5 | 219.42358 | 0.19092 | 0.57437 | -0.01316 |
| 5 | 261.44371 | 0.19092 | 0.57436 | -0.01316 |

Table 6.9: Characteristics of the putative ground states of compressed $LJ_{38}$ clusters for different values of $\mu_c$. We report the final energy and the BOPs $Q_4$, $Q_6$, and $\hat{W}_6$ of each structure.

computed BOP values indicate that the structures minimizing the modified energy function in Eq. (6.13) are almost identical to the optimal fcc configuration of the original, uncompressed problem. Doye and co-workers (Doye et al., 1999b) show that, among the different BOPs, $Q_4$ is best suited for discriminating between fcc and ico structures, with $Q_4(\mathbf{x}^{\text{ico}}) = 0$ and $Q_4(\mathbf{x}^{\text{fcc}}) = 0.1909$.

Table 6.10 summarizes our proposed specification for the tunable $LJ_{38}$ cluster benchmark. The bounds are far from being tight with respect to the optimal structure. Both $\mathbf{x}^{\text{ico}}$ and $\mathbf{x}^{\text{fcc}}$ would fit into the $[-2, 2]^n$ box. We propose the larger bounds for two reasons: First, we want to minimize effects from boundary handling techniques. Second, we want to test the capability of the optimization algorithm to cope with "uninformative" regions of parameter space. Enlarging the box adds plateau-like regions to the energy landscape because a particle that is far away from the cluster experiences only a small force that draws it toward the cluster.

| Problem | $LJ_{38}$ with compression |
|---|---|
| Runs per problem | 25 |
| $n$ | 108 |
| $\mu_c$ | $0\epsilon, 1\epsilon, \ldots, 5\epsilon$ |
| MAX_FES | $\sim 10^6 n$ |
| Termination | If FES = MAX_FES or $E_{LJ,\mu_c}(\mathbf{x}) \leq E_{LJ,\mu_c}(\mathbf{x}_{min}) + 10^{-6}$ |
| Initialization and bounds | Uniformly random in $[-4, 4]^n$ |

Table 6.10: Suggested benchmark settings for the tunable $LJ_{38}$ cluster test case.

We present numerical experiments with IPOP-CMA-ES using the specifications as outlined in Table 6.10. The LJ parameters $\sigma_{LJ}$ and $\epsilon$ are set to 1. We use standard IPOP-CMA-ES with $incPop = 1.25$. We test different initialization schemes that are inspired by Leary's big bang algorithm (Leary, 1997). Leary showed that a successful strategy for optimizing LJ clusters with $N < 55$ is to start from a dense initial random packing followed by a limited phase of steepest descent steps with fixed step size ("big bang" like). The resulting configurations are re-optimized by a first-order optimization method. We examine whether the initialization procedure of particle positions has an effect on the quality of the solutions found by IPOP-CMA-ES. We place the initial population mean uniformly at random in boxes of increasing size: $[-0.5, 0.5]^3$, $[-1.5, 1.5]^3$, and $[-3, 3]^3$ respectively. The initial step size $\sigma_{init}$ is set to 20% of the respective box length. We repeat the experiment 25 times for each box size and stop the optimization run whenever the population size exceeds the initial $\lambda$ by a factor of 100, i.e., after 21 restarts. This corresponds to MAX_FES $\approx 10^6 n$. We consider both the minimum and maximum compression factor $\mu_c = 0\epsilon$ and $\mu_c = 5\epsilon$.

Figure 6.15 summarizes the results. The experiments reveal that on the $LJ_{38}$ problem without compression none of the IPOP-CMA-ES runs reach the minimum energy configuration $\mathbf{x}^{fcc}$, nor the low-energy configuration $\mathbf{x}^{ico}$ (blue □ in Figure 6.15). With compression, however, all IPOP-CMA-ES runs find the globally optimal fcc configuration, independent of initialization. All local minima found on the problem without compression have icosahedral-like configurations with $Q_4$ values between 0.01 and 0.09. Runs that started in the largest box (blue •'s in Fig. 6.15) converged to minima that have the largest structural diversity of the ($Q_4 = 0.01 \ldots 0.045$), including the lowest found local minimum found at $E_{LJ} \approx -172.98\epsilon$. According to the $LJ_{38}$ disconnectivity graph constructed by Doye and Wales (Doye et al., 1999b) this minimum is the fourth lowest of all ico configurations and the fifth lowest among all configurations.

These results confirm the observations of Lunacek and co-workers (Lunacek et al., 2008). The failure of CMA-ES on the standard $LJ_{38}$ problem can be explained by the irresolvable trade-off for the optimal population size of CMA-ES on multi-funnel functions (Lunacek et al., 2008).

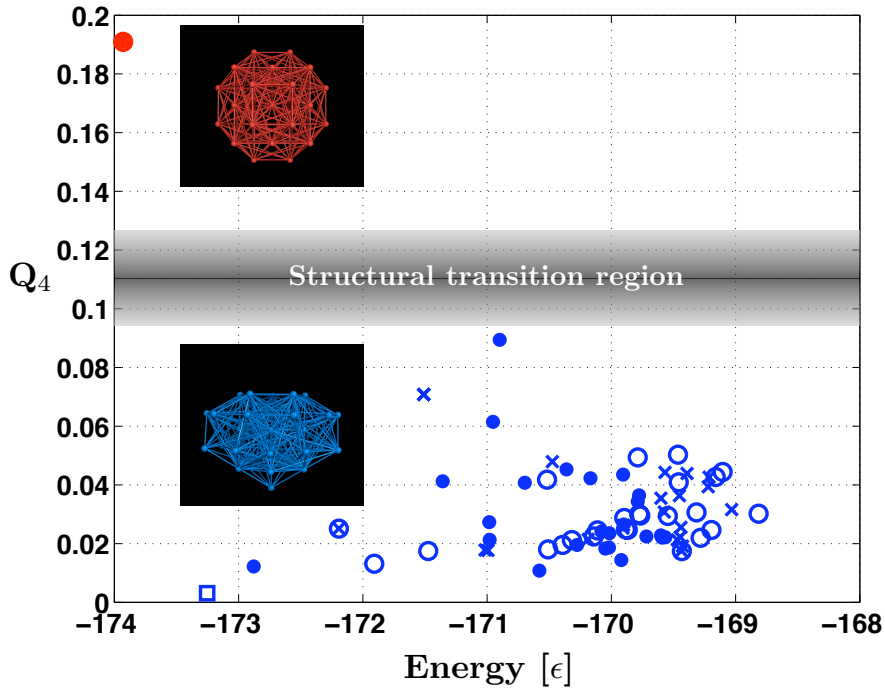



Figure 6.15: Bond-order parameter $Q_4$ vs. potential energy (in units of $\epsilon$) for all minima found by IPOP-CMA-ES. The blue data points show the $3 \cdot 25$ local minima found on the $LJ_{38}$ problem without compression with the initial $[-0.5, 0.5]^3$ box ($\times$), the $[-1.5, 1.5]^3$ box ($\circ$), and the $[-3, 3]^3$ box ($\bullet$). The single blue $\square$ in the bottom-left corner marks the lowest-energy icosahedral configuration. The shaded gray area is the structural transition region from ico to fcc symmetry (Doye et al., 1999b). The red data point in the upper-left corner corresponds to the global minimum ($E_{\mathrm{LJ}} = -173.9284\epsilon$), which was found by IPOP-CMA-ES in all $3 \cdot 25$ runs $\mu_{\mathrm{c}} = 5\epsilon$.

The larger the population size, the more likely it is that CMA-ES converges to the broadest, sub-optimal funnel. IPOP-CMA-ES robustly solves the problem when the PEL of $LJ_{38}$ is compressed to a single-funnel structure. Furthermore, we conclude that IPOP-CMA-ES is insensitive to the initialization procedure. A big bang-like initialization does not lead to an improved algorithmic performance.

To date, no gradient-free black-box optimization algorithm has been reported to solve the $LJ_{38}$ test case without compression. Since many real-world applications involve multi-funnel landscapes, we believe that the tunable $LJ_{38}$ problem with varying degree of compression presents a challenging test case for the black-box optimization community that might prove instrumental in the design and analysis of new search heuristics.

## 6.5 Alternative cluster benchmark problems

In the previous sections we have reviewed the physical and computational foundations of atomic cluster problems. We have presented two specific instances in detail: Cohn-Kumar clusters and Lennard-Jones clusters. It is clear that, depending on the pair potential and the space in which the atoms live, a large set of alternative benchmark problem sets could also be designed. We believe that *Morse clusters* and *minimum second-moment sphere packings* are of particular interest. In Morse clusters, the atoms interact via the Morse pair potential $u_{\text{Morse}}$ (Morse, 1929):

$$u_{\text{Morse}}(r_{ij}) = \epsilon\, e^{\rho(1-r_{ij}/r_{\text{e}})} \left( e^{\rho(1-r_{ij}/r_{\text{e}})} - 2 \right) , \tag{6.14}$$

where $r_{ij} = d_{\text{E}}(\mathbf{p}_i, \mathbf{p}_j)$. The parameter $\epsilon$ defines the well depth, $r_{\text{e}}$ is the pair equilibrium separation, and $\rho$ controls the "width" of the potential well. The smaller the value of $\rho$, the larger the potential well. Physically meaningful values are $\rho \in [3, 14]$ (Wales, 2005). It is well known that Morse cluster landscapes become increasingly rugged for larger $\rho$. For the 13-atom Morse cluster, a thorough characterization of the dependence of the landscape topography on $\rho$ can be found in (Cox et al., 2006). It is shown that the single-funnel character of the landscape is conserved across several $\rho$ values, but that the number of minima increases dramatically. It is, therefore, straightforward to use the Morse potential to construct black-box optimization test cases with a tunable degree of ruggedness, yet similar global topology. Putative ground states of Morse clusters for $N \leq 80$ and different $\rho$ values are reported in the Cambridge Cluster Database.

The minimum second-moment sphere packing problem exhibits further characteristics that are not covered by the present benchmarks. This problem considers the arrangement of finitely many non-overlapping, identical hard spheres that fill the 3D space. Johannes Kepler conjectured in 1611 that the optimal arrangement of *infinitely* many hard spheres (in fact, the original problem was stated with canon balls) is achieved by an fcc packing. A proof for this conjecture was presented in 1998 by Thomas Hales. His proof by exhaustion is considered almost certainly correct (see (Hales, 2005) for the final publication). When the number $N$ of spheres is, however, finite, there is no unique statement of the problem. Sloane and co-workers were the first to investigate the nowadays most common formulation (Sloane et al., 1995): finding the configuration of non-overlapping spheres that has the smallest second moment of the positions about the center of mass. This defines a non-convex objective function with quadratic constraints. This objective function is, in fact, identical to the compression penalty $Q_{\text{c}}$ in the $\text{LJ}_{38}$ test case. The hard-sphere constraints, however, turn the minimum second-moment sphere packing problem into a discontinuous problem where derivatives do not exist. Sloane and co-workers applied a variety of methods to this problem, ranging from direct search heuristics (Simulated Annealing) to complete enumeration. In the original article, they presented putative optimal configurations up to $N = 32$. Putative optimal configurations up to $N = 99$ are listed in (Sloane et al., 1997). Recently, Arkus and co-workers provided a geometric enumeration approach that confirms the optimality of finite sphere packings up to $N = 10$ (Arkus et al., 2009). These optimal structures and their expected formation probabilities were also confirmed in recent experiments using polystyrene particles immersed in a

mixture of water and micro-gel particles (Meng et al., 2010). Both the theoretical and experimental studies suggest that the energy landscapes of sphere packing problems are not strongly funneled, but contain distinct local minima that are separated by large barriers. We expect that confirming or improving the currently known putative optimal finite sphere packings is a formidable challenge for black-box optimization methods.

## 6.6 Conclusions

Finding the optimal spatial arrangement of atoms that minimizes the potential energy of a cluster system constitutes a promising problem class for continuous black-box optimization benchmarking. We presented several atomic cluster problems and analyzed their energy landscapes and putative optima. We focused on Cohn-Kumar clusters and Lennard-Jones clusters, whose energies are given by sums over distance-dependent pair-wise potentials.

We have shown that Cohn-Kumar clusters have smooth, globally convex landscapes with a single or few minima. On CK cluster instances for which proven ground states are known, we compared the performance of a restart Nelder-Mead simplex algorithm with that of IPOP-CMA-ES. IPOP-CMA-ES outperformed Nelder-Mead in terms of robustness, speed, and solution quality. For all other CK clusters up to $N = 25$ we found putative global minima and several low-energy local optima from extensive numerical simulations. This provides the necessary information for a benchmark suite in up to $n = 50$ dimensions.

The presented Lennard-Jones cluster instances are known to exhibit rugged single-funnel topologies as well as *tunable* double-funnel topologies. IPOP-CMA-ES was able to identify putative ground states for LJ clusters up to $N = 19$. We further proposed the 38-atom LJ cluster with compression as a benchmark to assess the sensitivity of search algorithms with respect to landscape topology.

All cluster problems possess isospectral symmetry as a novel characteristic that is not covered by the test functions in the current black-box benchmark suites. Cluster problems hence provide a means of determining whether and how well black-box algorithms can cope with this problem feature. We suggest using bond-order parameters as symmetry-invariant measures to characterize and compare structures. Search trajectories of black-box optimizers can also be conveniently represented using these parameters. We believe that atomic cluster problems should be included in future black-box benchmark studies in order to better assess the efficacy, efficiency, and generality of search heuristics.

# 7

# Analysis of Linear Chain Landscapes

"Uh, oh, Spaghetti-O's!"
Homer Simpson, in: The Simpsons, Homer to the Max, Episode no. 216, 1999

Another important class of molecular systems that can be analyzed from an energy landscape perspective are *chain molecules*. In chain molecules, a collection of atoms is connected by molecular bonds to a three-dimensional, unbranched linear chain. Such molecules are ubiquitous in nature. Important instances of biological chain molecules (or bio-polymers) are DNA, RNA, and proteins. Since the pioneering works of Flory (Flory, 1953, 1969), simplified model chains, such as the freely jointed or ideal Random Walk chain or the self-avoiding random walk, have been fundamental for statistical considerations of polymers. They provide the theoretical basis for more complex (bio-)polymer models and proteins (Cantor and Schimmel, 1980). For the last three decades, such complex chain molecules can also be studied by computational techniques. Nowadays the energy landscape of small globular proteins can be regularly explored on a standard computer. Processes like protein folding, where a "spaghetto-like" unfolded chain turns into a unique, regularly structured folded shape, can be simulated up to the millisecond time scale. The potential energy of a chain is modeled analogously to the atomic systems considered in the previous chapter. Carefully parametrized atomic pairwise potentials are used in order to reproduce measured thermodynamic properties of the molecules. The resulting *force fields* enable the use of the Molecular Dynamics (MD) simulation method to explore the energy landscape. There, the molecular system is evolved over time following Newton's equation of motion. An alternative way to sample the energy landscape is provided by MCMC with specific molecular move sets. The typical data produced by these simulations are large lists of molecular conformations that either constitute a

trajectory over the landscape or a representative sample from the whole landscape domain.

A fundamental prerequisite for studying and interpreting such conformational data sets is the use of a proper distance measure. The advent of efficient algorithms for determining the minimal atom-positional Root Mean Square Distance (or Deviation) (RMSD) (McLachlan, 1972, 1979; Kabsch, 1976, 1978) between two chain molecules turned RMSD into the prevalent distance measure in the molecular sciences. Countless scientific publications use this measure for structure comparison. Despite its universal applicability, mathematical results on the properties of RMSD in the context of chain molecules are sparse. The statistical distributions of RMSD values have been probed for ideal Random Walk ensembles by McLachlan (McLachlan, 1984) and for more complex polymer models and proteins by Reva and co-workers (Reva et al., 1998) and by Sullivan and Kuntz (Sullivan and Kuntz, 2001; Sullivan et al., 2003; Sullivan and Kuntz, 2004). Maiorov and Crippen elucidated that for globular proteins the statistical significance of RMSD values depends on the length of the chain (Maiorov and Crippen, 1994, 1995), i.e., absolute RMSD values have to be interpreted relative to the length of the molecule.

From a landscape perspective it is also important to understand what neighborhood structure the RMSD metric induces. In the standard neighborhood definition a structure is a neighbor of some other structure if their pairwise RMSD is smaller than a given cutoff value $R_c$. Imagine now a large random ensemble of linear chains that has been created in the absence of any potential energy terms, i.e., in a flat energy landscape. When probing such a data set with RMSD, we might expect that all chains have approximately the same number of neighbors. However, we will show that this expectation is false for the simplest class of linear chains, the Random Walk model even in the limit case. We quantify the amount of inhomogeneity in the neighborhood density of short Random Walks and identify two limiting configurations: (i) the densest (or most probable) structures and (ii) the barycentric structure (Section 7.2). In this chapter, we use the terms shape probability (density) and neighborhood density interchangeably.

Furthermore, we conjecture an upper bound for the RMSD between linear chains. This bound also defines the diameter of the conformational landscape spanned by a pair of extremal structures (Section 7.3). The results presented in Section 7.2 have been obtained in close collaboration with Dr. Philippe Hünenberger and Dr. Bojan Žagrović. The interested reader is referred to the corresponding article (Müller et al., 2009) for a more detailed presentation.

## 7.1 Linear chains: Conformation space and distance definition

We first introduce the Random Walk (RW) and Self-avoiding random walk (SAW) linear chain models that define the conformational landscape domain $\mathcal{X}$. We then present the RMSD metric that serves as the distance measure $d$ between linear chains. We analyze the configuration space in the absence of any force or energy, thus $f \equiv 0$.

### 7.1.1 Random Walks and Self-avoiding Walks

The simplest linear chain model is the Random Walk model. A walk of length $N \geq 3$ and step size $b$ corresponds to the path obtained (in three-dimensional Cartesian space) by starting at some origin and taking $N-1$ successive straight steps of equal length $b$ in arbitrary directions, as illustrated in Fig. 7.1. The $N$ points along such a path are referred to as beads and the corresponding steps as bonds or links. Walks defined in this way are: (i) unbranched (linear topology); (ii) non-self-avoiding (beads may be positioned arbitrarily close to each other, except for consecutive ones) ; (iii) oriented (a walk is distinct from its reverse walk, as defined by taking the beads in reverse order).

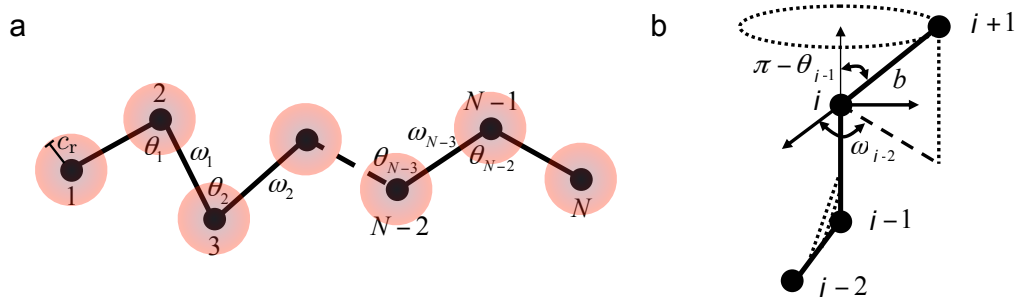



Figure 7.1: a. Definition of the angles $\theta_n$ and dihedral angles $\omega_n$ characterizing an anchored walk of length $N$. $c_r$ is the radius of a sphere that defines the excluded volume for a self-avoiding walk. b. Illustration of the conversion $\mathbf{q} \to \mathbf{r}(\mathbf{q})$ from an internal coordinate vector $\mathbf{q}$ to the corresponding Cartesian coordinate vector $\mathbf{r}$ via trigonometry.

A walk can be entirely specified by the $3N$-dimensional vector $\mathbf{r} \doteq \{\mathbf{p}_i, i = 1, \ldots, N\} = \{r_\alpha | \alpha =, \ldots, 3N\}$, where $\mathbf{p}_i$ is the Cartesian coordinate vector of bead i and $r_\alpha$ a single Cartesian coordinate within $\mathbf{r}$. To avoid the redundancy of walks that can be superimposed by trivial rigid-body translation and rotation, it is convenient to define *anchored* walks as the walks satisfying the six additional constraints $r_\alpha = 0$ for $\alpha = 1, 2, 3, 5, 6, 9$, along with $r_4 = b$ and $(r_7 - b)^2 + r_8^2 = b^2$. In other words, for an anchored walk, the $\mathbf{p}_1$ bead is placed at the origin, the $\mathbf{p}_1 - \mathbf{p}_2$ bond aligned along the $x$-axis, and the $\mathbf{p}_2 - \mathbf{p}_3$ bond contained in the $xy$-plane, which uniquely defines the overall (rigid-body) position and orientation of the walk. Due to the $N-1$ bond plus six rigid-body constraints, the space $\mathcal{C}_N$ spanned by the Cartesian coordinate vectors $\mathbf{r}$ associated with all anchored walks of length $N$ represents an $n$-dimensional hypersurface within $\mathbb{R}^{3N}$, where $n = 2N - 5$. Alternatively, an anchored walk can also be entirely specified by an $n$-dimensional internal coordinate vector $\mathbf{q} \doteq \{q_1 = 1, \ldots, n\}$, where $q_i$ $(i = 1, \ldots, N-2)$ is the cosine of the angle $\theta_i$ formed by $\mathbf{p}_i - \mathbf{p}_{i+1} - \mathbf{p}_{i+2}$ and $q_{i+N-2}$ $(i = 1, \ldots, N-3)$ is the dihedral angle $\omega_i$ formed by $\mathbf{p}_i - \mathbf{p}_{i+1} - \mathbf{p}_{i+2} - \mathbf{p}_{i+3}$ (oriented and measured in radians), see Fig. 7.1. This can be written as:

$$\mathbf{q} \doteq \{\{\cos(\theta_i) | i = 1, \ldots, N-2\}, \{\omega_i = 1, \ldots, N-3\}\} \; . \tag{7.1}$$

The $N-2$ angle-cosine coordinates are non-periodic and bounded to the range $[-1; 1]$. The

$N-3$ dihedral-angle coordinates are periodic and chosen here (by convention) within the range $]-\pi;\pi]$. The $n$-dimensional space $\mathcal{Q}_N$ spanned by the internal coordinate vectors $\mathbf{q}$ corresponding to all anchored walks of length $N$ is thus compact (no "holes") and bounded, with a finite volume $V_{\mathcal{Q}_N}$ given by

$$V_{\mathcal{Q}_N} \doteq \int_{\mathcal{Q}_N} d^n\mathbf{q} = 2^{N-2}(2\pi)^{N-3} = 2^n\pi^{(n-1)/2} \ . \tag{7.2}$$

Note that if any bond angle $\theta_n$ of the walk is equal to 0 or $\pi$, the preceding and succeeding dihedral angles ($\omega_{i-1}$ and $\omega_i$) are undefined and need to be replaced by a single dihedral angle $\mathbf{p}_{i-1} - \mathbf{p}_i - \mathbf{p}_{i+2} - \mathbf{p}_{i+3}$ (assuming that $\theta_{i-1}$ and $\theta_{i+1}$ themselves differ from 0 and $\pi$). This special handling turns out to be necessary in Section 7.3.

The mapping $\mathbf{q} \to \mathbf{r}(\mathbf{q})$ of an anchored walk from $\mathcal{Q}_N$ to $\mathcal{C}_N$, as well as the reverse mapping $\mathbf{r} \to \mathbf{q}(\mathbf{r})$ from $\mathcal{C}_N$ to $\mathcal{Q}_N$, are defined and unique (except for walks involving one or more bond angles equal to 0 or $\pi$), as well as continuous. Both transformations can be performed using straightforward trigonometry, as illustrated in Fig. 7.1. Because $\mathcal{Q}_N$ is compact and bounded (with a finite volume $V_{\mathcal{Q}_N}$), the uniqueness and continuity of the transformation implies that the hypersurface $\mathcal{C}_N$ (within $\mathbb{R}^{3N}$) is also compact and bounded, with a finite area $A_{\mathcal{C}_N}$. The relationship between the two spaces is illustrated schematically in Fig. 7.2.

The *random walk ensemble* $\mathcal{W}_N$ is defined as an infinite ensemble of anchored walks of length $N$ with a homogeneous (normalized) probability distribution $p_N$ over $\mathcal{Q}_N$:

$$p_N(\mathbf{q}) = V_{\mathcal{Q}_N}^{-1} \quad \text{so that} \quad \int_{\mathcal{Q}_N} d^n\mathbf{q}\, p_N(\mathbf{q}) = 1 \, . \tag{7.3}$$

It is easily seen that $\mathcal{W}_N$ can be generated by taking (an infinite number of) walks in $\mathcal{C}_N$ for which each successive step of length $b$ is taken in a random (i.e. isotropically distributed) direction, keeping in mind the six constraints imposed to the Cartesian coordinate components of the first three beads.

Self-avoidance in RW's is modeled by considering a sphere of radius $c_\mathrm{r}$ centered at each bead position that defines an excluded volume. This volume cannot be penetrated by any other excluded-volume sphere (see Fig. 7.1). Such a self-avoidance constraint cannot be defined in internal coordinates because of its non-locality. Hence the generation of an ensemble of SAW's is more involved. The simplest approach is to generate a RW ensemble and remove all structures that do not satisfy the self-avoidance constraint.

### 7.1.2 RMSD as distance metric for linear chains

We consider the root-mean-square atomic positional deviation (RMSD) $D$ after least-squares roto-translational fitting as pairwise metric for neighborhood definition. For a given reference structure $\mathbf{r}$ and a given compared structure $\mathbf{r}'$ , $D$ is the metric representing the scalar distance between two associated 3N-dimensional vectors $\mathbf{s}(\mathbf{r})$ and $\mathbf{s}'(\mathbf{r},\mathbf{r}')$ defined by $\mathbf{s}(\mathbf{r}) = N^{-1/2}\mathbf{r}$ and

$$\mathbf{s}'_i(\mathbf{r},\mathbf{r}') \doteq N^{-1/2}\,(\mathbf{R}\mathbf{p}'_i + \mathbf{t}) \quad , \quad i = 1,\ldots,N \ , \tag{7.4}$$

namely

$$D(\mathbf{r},\mathbf{r}') \doteq |\mathbf{s}'(\mathbf{r},\mathbf{r}') - \mathbf{s}(\mathbf{r})| = \left\{ N^{-1} \sum_{i=1}^{N} \left[\mathbf{R}\mathbf{p}_i' + \mathbf{t} - \mathbf{r}_i\right]^2 \right\}^{1/2} . \tag{7.5}$$

Here, $\mathbf{R}$ and $\mathbf{t}$ denote the three-dimensional rotation matrix (three degrees of freedom) and translation vector (three degrees of freedom) leading to the minimum value of $D$ for the given pair of structures. It can be shown that $D$ satisfies all properties of a metric in the mathematical sense (Kaindl and Steipe, 1997; Steipe, 2002a,b) (positivity: $D(\mathbf{r},\mathbf{r}) = 0$ and $D(\mathbf{r},\mathbf{r}') > 0 \; \forall \; \mathbf{r}' \neq \mathbf{r}$; symmetry: $D(\mathbf{r},\mathbf{r}') = D(\mathbf{r}',\mathbf{r}) \; \forall \; \mathbf{r},\mathbf{r}'$; triangle inequality: $D(\mathbf{r},\mathbf{r}'') \leq D(\mathbf{r},\mathbf{r}') + D(\mathbf{r}',\mathbf{r}'') \; \forall \; \mathbf{r},\mathbf{r}',\mathbf{r}''$). A number of alternative exact procedures for determining $\mathbf{R}$ and $\mathbf{t}$ from $\mathbf{r}$ and $\mathbf{r}'$ have been proposed in the literature. Kabsch's algorithm uses Singular Value Decomposition (SVD) (Kabsch, 1976, 1978) while Horn proposes quaternions (Horn, 1987; Horn et al., 1988). For a given reference structure $\mathbf{r}$, the $n$-dimensional hypersurface (within $\mathbb{R}^{3N}$) containing the vectors $\mathbf{s}'(\mathbf{r},\mathbf{r}')$ associated with all anchored walks $\mathbf{r}'$ of length $N$ will be noted $\mathcal{R}_N(\mathbf{r})$. This hypersurface contains the $3N$-dimensional Cartesian coordinate vectors (amplified by $N^{-1/2}$) of all anchored walks after least-squares roto-translational fitting onto $\mathbf{r}$. Because $\mathcal{Q}_N$ is compact and bounded (with a finite volume $V_{\mathcal{Q}_N}$), for any $\mathbf{r}$, the hypersurface $\mathcal{R}_N(\mathbf{r})$ (within $\mathbb{R}^{3N}$) is also compact and bounded, with a finite area $A_{\mathcal{R}_N(\mathbf{r})}$. The hypersurface $\mathcal{R}_N(\mathbf{r})$ associated with the $D$ metric depends on the choice of the reference structure $\mathbf{r}$. However, if only distances between very close structures are of interest, it is possible to piece the $\mathcal{R}_N(\mathbf{r})$ hypersurfaces together from single hypersurfaces $\tilde{\mathcal{R}}_N$ with metrics $\tilde{D}$ that are locally equivalent to $D$, i.e., satisfying

$$\lim_{D(\mathbf{r},\mathbf{r}')\to 0} [D(\mathbf{r},\mathbf{r}') - \tilde{D}(\mathbf{r},\mathbf{r}')] = 0 \; . \tag{7.6}$$

This can be done by introducing a regular paving of $\mathcal{Q}_N$ using $G$ grid cells centered at grid points $\{\mathbf{q}_k | k = 1,\ldots,G\}$. The hypersurface $\tilde{\mathcal{R}}_N$ is then defined as

$$\tilde{\mathcal{R}}_N \doteq \lim_{G\to\infty} \cup_{k=1}^{G} \mathcal{R}_{N,k} \; , \tag{7.7}$$

where $\mathcal{R}_{N,k}$ denotes the portion of $\mathcal{R}_N(\mathbf{r}(\mathbf{q}_k))$ corresponding to all structures contained within the grid cell $k$. The metric $\tilde{D}$ in $\tilde{\mathcal{R}}_N$ is the scalar distance

$$\tilde{D}(\mathbf{r},\mathbf{r}') \doteq \lim_{G\to\infty} |\tilde{\mathbf{s}}_G'(\mathbf{r}') - \tilde{\mathbf{s}}_G(\mathbf{r})| \; , \tag{7.8}$$

where $\tilde{\mathbf{s}}_G(\mathbf{r})$ and $\tilde{\mathbf{s}}_G'(\mathbf{r}')$ represent the $3N$-dimensional Cartesian coordinate vectors (amplified by $N^{-1/2}$) of the two anchored walks after least-squares roto-translational fitting onto the structures associated with the two respective closest grid points (for a given $G$). Note that unlike $\mathbf{s}'$, $\tilde{\mathbf{s}}_G'$ only depends on $\mathbf{r}'$ (not on $\mathbf{r}$).

When comparing structures at a finite distance $D$, $\tilde{\mathcal{R}}_N$ is essentially equivalent to $\mathcal{C}_N$ amplified by $N^{-1/2}$, and $\tilde{D}$ represents the RMSD-like distance between two anchored walks without any roto-translational fitting. Howewer, at the local level, $\tilde{\mathcal{R}}_N$ is not equivalent to $\mathcal{C}_N$ scaled by $N^{-1/2}$. For any finite $G$, the patched hypersurface (union of the $\mathcal{R}_{N,k}$) is discontinuous at the grid-cell boundaries, and this discontinuity survives in $\tilde{\mathcal{R}}_N$ at the infinitesimal (local) level when taking the limit $G \to \infty$. It is easily seen that Eq. (7.6) holds, provided the limit

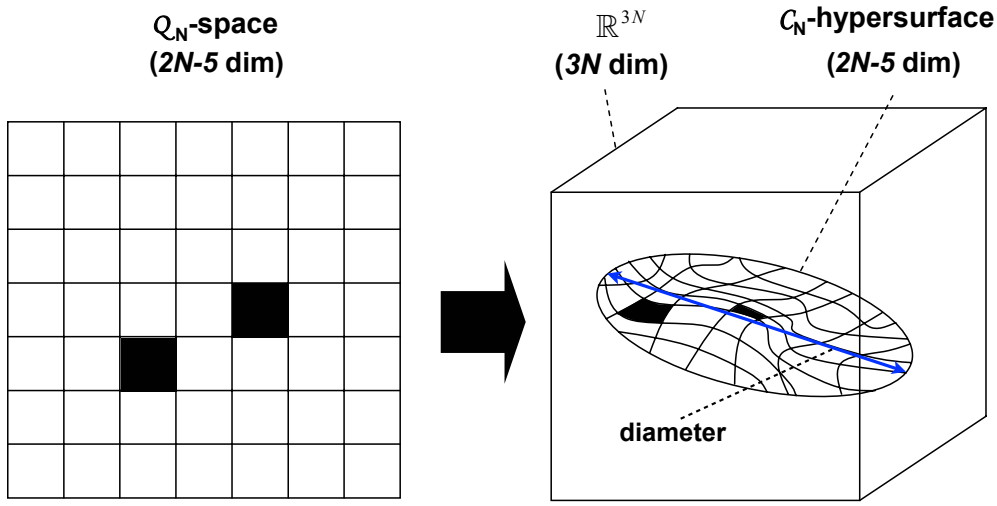



Figure 7.2: Schematic representation of the coordinate transformation from $\mathcal{Q}_N$ ($n$-dimensional internal coordinate space, with $M = 2N - 5$) to $\mathcal{C}_N$ ($n$-dimensional hypersurface within the entire Cartesian coordinate space $\mathbb{R}^{3N}$) for anchored walks of length $N$. $\mathcal{Q}_N$ is bounded and of finite volume $V_{\mathcal{Q}_N}$ (Eq. (7.2)). $\mathcal{C}_N$ is also bounded and of finite area $A_{\mathcal{C}_N}$. Note that the $N - 3$ dihedral-angle coordinates within $\mathbf{q}$ are actually periodic (i.e., bounded only by the definition of a reference interval). For simplicity, this periodicity (i.e., the "folding" of a part of the boundary of $\mathcal{Q}_N$ or $\mathcal{C}_N$ on itself) is not represented in the figure. The same drawing could also illustrate the coordinate transformation from $\mathcal{Q}_N$ to $\tilde{\mathcal{R}}_N$ ($n$ dimensions within $\mathbb{R}^{3N}$), the hypersurface associated with the local RMSD metric $\tilde{R}$ of Eq. (7.8). In this case, the fact that two infinitesimal volumes of $\mathcal{Q}_N$ transform to patches of different areas of $\tilde{\mathcal{R}}_N$ indicates that the corresponding shapes have different local probabilities $P_N(\mathbf{q}, 0)$; (Eq. (7.13) and Eq. (7.14)). In the present case, the patch on the right is representative of a (locally) more probable shape compared to the one on the left.

$D(\mathbf{r}, \mathbf{r}') \to 0$ in this equation is taken before the limit $G \to \infty$ in Eq. (7.8), provided that the distance between the two compared structures remains infinitesimal compared to the grid spacing, even when taking the latter toward zero.

Because $\mathcal{Q}_N$ is compact and bounded (with a finite volume $V_{\mathcal{Q}_N}$), the hypersurface $\tilde{\mathcal{R}}_N$ (within $\mathbb{R}^{3N}$) is also compact and bounded, with a finite area $A_{\tilde{\mathcal{R}}_N}$. For the above-mentioned reasons, however, $A_{\tilde{\mathcal{R}}_N}$ is not equal to $N^{-1/2} A_{\mathcal{C}_N}$. The drawing in Fig. 7.1 could thus also apply to the relationship between $\mathcal{Q}_N$ and $\tilde{\mathcal{R}}_N$, keeping in mind the peculiar local properties of the latter hypersurface.

The RMSD metric as a measure of structural dissimilarity has the convenient properties that it is: (i) independent of the (rigid-body) relative positioning and orientation of the two compared structures, and (ii) unaffected by performing a mirror symmetry, a central inversion or an atom renumbering on the two compared structures. Note, however, that since the present walks are oriented, the distance between a walk and its reverse walk is in general not zero. Although the RMSD metric is probably the most appropriate one to match our visual intuition concerning structural difference, it is not the only possible choice. For example, a distance-matrix root-mean-square difference (Zagrovic et al., 2002; de Araujo et al., 2008) could be

more appropriate to match our expectations concerning structure-related energy differences for systems where the dominant interactions correlate with pairwise interatomic distances. In contrast, an RMSD without roto-translational fitting would represent a poor measure, in the intuitive sense, of the structural dissimilarity between two walks, because the anchoring of the walks in the $\mathcal{C}_N$ space (performed here on the first three beads) is arbitrary. This would mean in particular that: (i) differences in the first angles and dihedral angles along the walk will have more influence on the metric compared to corresponding differences at the end of the walk; (ii) the distance between two walks would not be equal to the distance between the two corresponding reverse walks. A root-mean-square difference in internal coordinates would also represent a poor metric for structure comparison, in particular because: (i) differences in the angles and dihedral angles would be equally weighted along the chain, although the central ones are intuitively expected to have more impact on the overall shape compared to the terminal ones; (ii) dihedral angles are periodic variables, so that the resulting measure would depend on the arbitrary choice of a reference interval for the dihedral angles.

## 7.2 The neighborhood density of Random Walk chains

The neighborhood of any linear chain is defined here as the collection of all structures for which the RMSD to a reference (or central) linear chain is below a given cutoff value $R_\mathrm{c}$. We call this collection of structures also a "shape" or "state". This definition implies in particular that: (i) every structure can be used to define a shape; (ii) different shapes may be overlapping in terms of the structures they encompass, i.e., individual structures are not necessarily only associated to a single shape. For simplicity, the central structure of a shape will be noted $\mathbf{q}$, i.e., as an internal coordinate vector of $\mathcal{Q}_N$ (with the corresponding Cartesian coordinate vector in $\mathcal{C}_N$ noted $\mathbf{r} \doteq \mathbf{r}(\mathbf{q})$), and the shape of which $\mathbf{q}$ is the central structure will be loosely referred to as the shape $\mathbf{q}$.

The problem considered here is to determine how the homogeneous internal-coordinate probability distribution $p_N(\mathbf{q})$, defined in Eq. (7.3) and associated with individual structures in the random walk ensemble $\mathcal{W}_N$, induces a corresponding neighborhood or shape probability distribution $P_N(\mathbf{q}, R_\mathrm{c})$ in structure space. This shape probability distribution will be normalized to $V_{\mathcal{Q}_N}$ (Eq. (7.2)) rather than to unity:

$$\int_{\mathcal{Q}_N} d^n\mathbf{q}\, P_N(\mathbf{q}, R_\mathrm{c}) = V_{\mathcal{Q}_N} \tag{7.9}$$

This permits an immediate interpretation of $P_N(\mathbf{q}, R_\mathrm{c})$ as the probability that an arbitrary random walk from $\mathcal{W}_N$ belongs to the specific shape $\mathbf{q}$, relative to the average of this probability over all possible shapes. For example, a value of 1.1 for $P_N(\mathbf{q}, R_\mathrm{c})$ indicates that, for the given cutoff $R_\mathrm{c}$, shape q is 10% more likely to encompass an arbitrary random walk, compared to any shape of $\mathcal{Q}_N$ taken at random. The probability $P_N(\mathbf{q}, R_\mathrm{c})$ can also be interpreted as the sub-volume of $\mathcal{Q}_N$ spanned by the specific shape $\mathbf{q}$, relative to the average of this sub-volume over all possible shapes. For example, a value of 1.1 for $P_N(\mathbf{q}, R_\mathrm{c})$ also indicates that, for the given cutoff $R_\mathrm{c}$, the neighborhood of structure $\mathbf{q}$ spans a 10% larger sub-volume of $\mathcal{Q}_N$ compared to the neighborhood of any shape of $\mathcal{Q}_N$ taken at random. $P_N$ is thus a measure of the average density of random walks in the neighborhood of structure $\mathbf{q}$ (i.e., within the shape

$\mathbf{q}$). When comparing two shapes, the ratio of the corresponding $P_N$ values indicates how much more likely one of them is compared to the other. Finally, the chosen normalization implies that if all shapes were equiprobable, $P_N$ would be uniformly one over $\mathcal{Q}_N$. In the following discussion, the cases of a finite versus an infinitesimal cutoff $R_\mathrm{c}$ are discussed consecutively.

**The finite cutoff case.** Consider first the case of a finite cutoff. For the RMSD metric, a given shape $\mathbf{q}$ with $\mathbf{r} \doteq \mathbf{r}(\mathbf{q})$ can be assigned a weight $\Omega_N(\mathbf{q}, R_\mathrm{c})$ defined by the sub-volume of $\mathcal{Q}_N$ mapping to the region of the hypersurface $\mathcal{R}_N(\mathbf{r})$ enclosed within a $3N$-dimensional hypersphere of radius $R_\mathrm{c}$ centered at $\mathbf{s}(\mathbf{r})$:

$$\Omega_N(\mathbf{q}, R_\mathrm{c}) \doteq \int_{\mathcal{Q}_N} d^M\mathbf{q}' \; \Theta(R_\mathrm{c} - |\mathbf{s}(\mathbf{r}(\mathbf{q}), \mathbf{r}'(\mathbf{q}')) - \mathbf{s}(\mathbf{r}(\mathbf{q}))|) \; , \tag{7.10}$$

where $\Theta$ is the Heaviside function. The corresponding shape probability density, normalized as defined in Eq. (7.9) may then be written as:

$$P_N(\mathbf{q}, R_\mathrm{c}) \doteq \frac{V_{\mathcal{Q}_N} \Omega_N(\mathbf{q}, R_\mathrm{c})}{\int_{\mathcal{Q}_N} d^n\mathbf{q}\, \Omega_N(\mathbf{q}, R_\mathrm{c})} \; . \tag{7.11}$$

Another quantity of interest is the fractional coverage function $f_N(\mathbf{q}, R_\mathrm{c})$, defined as the fraction of $\mathcal{Q}_N$ covered by $\Omega_N(\mathbf{q}, R_\mathrm{c})$:

$$f_N(\mathbf{q}, R_\mathrm{c}) \doteq V_{\mathcal{Q}_N}^{-1} \, \Omega_N(\mathbf{q}, R_\mathrm{c}) \; . \tag{7.12}$$

For a given value of $N$, the function $f_N(\mathbf{q}, R_\mathrm{c})$ is expected to present three regimes depending on the choice of $R_\mathrm{c}$: (i) For $R_\mathrm{c}$ below some threshold $R_N^*$, all shapes will only encompass a part of $\mathcal{Q}_N$, i.e., $f_N < 1$ for all $\mathbf{q}$; (ii) For $R_\mathrm{c}$ above some threshold value $R_N^{**} > R_N^*$, all shapes will encompass the entire extent of $\mathcal{Q}_N$, i.e., $f_N = 1$ (and, consequently, $P_N = 1$) for all $\mathbf{q}$ . This threshold is the *diameter* of the conformational space, which will be further discussed in Section 7.3; (iii) For intermediate values $R_N^* \leq R_\mathrm{c} \leq R_N^{**}$, a single shape ($R_\mathrm{c} = R_N^*$), and then an increasingly large set of shapes ($R_\mathrm{c} > R_N^*$), will extend over the entire $\mathcal{Q}_N$ ( i.e. $f_N = 1$), the other shapes still being characterized by $f_N < 1$. The single shape $\mathbf{q}_N^*$ (it can also be a few symmetry-related shapes) for which $f_N(\mathbf{q}_N^*, R_N^*) = 1$ has a special meaning. It is the shape that can encompass the entire extent of its $\mathcal{R}_N(\mathbf{r})$ hypersurface for the smallest possible value of the cutoff distance. For this reason, $\mathbf{q}_N^*$ will be referred to as the *barycentric* shape of $\mathcal{Q}_N$.

**The infinitesimal cutoff case.** Consider next the case of an infinitesimal cutoff, where $R_\mathrm{c} \to 0$ (which will loosely be written as $R_\mathrm{c} = 0$). In this case, $P_N(\mathbf{q}, 0)$ probes the local density of random walks within a shape of infinitesimal size centered at $\mathbf{q}$. Because distances within an infinitesimal shape are infinitesimal, and due to Eq. (7.6), it is possible to work here in the patched hypersurface $\tilde{\mathcal{R}}_N$ (Eq. (7.7)) rather than in the individual hypersurfaces $\mathcal{R}_N(\mathbf{r})$.

For the RMSD metric, if an infinitesimal volume element $d^n\mathbf{q}$ around $\mathbf{q}$ in $\mathcal{Q}_N$ maps to a corresponding infinitesimal hypersurface element $d^n\tilde{\Sigma}_N$ of $\tilde{\mathcal{R}}_N$ around $\mathbf{s}(\mathbf{r})$, the shape can be given a weight $\Gamma_N(\mathbf{q})$ defined by

$$\Gamma_N(\mathbf{q}) \doteq \frac{d^n\mathbf{q}}{d^n\tilde{\Sigma}_N} \; . \tag{7.13}$$

Intuitively, for a given $d^n\mathbf{q}$ around a central structure $\mathbf{q}$, a large $\Gamma_N$ (small $d^n\tilde{\Sigma}_N$) indicates that the random walks within $d^n\mathbf{q}$ are more densely packed in $\tilde{\mathcal{R}}_N$ around the central structure, i.e., the corresponding shape is more "likely". A small $\Gamma_N$ indicates that these walks are more widely spread and hence this shape is less likely. This correspondence is illustrated schematically in Fig. 7.2. The ratio $\Gamma_N$ represents the inverse of (the absolute value of) a Jacobian determinant of a special kind, which associates infinitesimal variations in an $n$-dimensional space ($\mathcal{Q}_N$) to corresponding variations in a $3N$-dimensional space ($\mathbb{R}^{3N}$) that are constrained to a $n$-dimensional hypersurface ($\tilde{\mathcal{R}}_N$). This Jacobian determinant is that of a $3N$-dimensional matrix containing in its first $n$ lines the variations $ds_\alpha/dq_i$ with $i = 1, \ldots, n$ and $\alpha = 1, \ldots, 3N$, and in its $3N - n = N + 5$ last lines, the coefficients of a set of $3N$-dimensional unit vectors that are orthogonal to those in the first $n$ lines as well as to each other. The corresponding local shape probability density, normalized as defined by Eq. (7.9), may then be written:

$$P_N(\mathbf{q}, 0) \doteq I_N^{-1}\, V_{\mathcal{Q}_N}^{-1}\, \Gamma_N(\mathbf{q}) \tag{7.14}$$

where $I_N$ is the average of $\Gamma_N$ over all shapes of $\mathcal{Q}_N$ divided by the volume $V_{\mathcal{Q}_N}$:

$$I_N \doteq V_{\mathcal{Q}_N}^{-2} \int_{\mathcal{Q}_N} d^n\mathbf{q}\, \Gamma_N(\mathbf{q}) \ . \tag{7.15}$$

The single shape $\mathbf{q}_N^{\#}$ (or a few symmetry-related shapes) that maximizes $P_N(\mathbf{q}, 0)$ over $\mathcal{Q}_N$ has a special meaning. It corresponds to the structure that has the highest density of random walks in its infinitesimal neighborhood. For this reason, $\mathbf{q}_N^{\#}$ will be referred to as the "densest" shape of $\mathcal{W}_N$. Finally, based on Eq. (7.13), the area $A_{\tilde{\mathcal{R}}_N}$ of $\tilde{\mathcal{R}}_N$ can be evaluated as:

$$A_{\tilde{\mathcal{R}}_N} = \int_{\mathcal{Q}_N} d^n\mathbf{q}\, \Gamma_N^{-1}(\mathbf{q}) \ . \tag{7.16}$$

Note that the above approach is not applicable to walks containing one or more bond angles equal to 0 or $\pi$, since these cannot be unambiguously represented in $\mathcal{Q}_N$. However, because they possess fewer than $n$ degrees of freedom, it is easily seen that they are characterized by a vanishing local shape probability density $P_N(\mathbf{q}, 0) = 0$.

It is important to realize that the weight $\Gamma_N$ ($R_c \to 0$; Eq. (7.13)) differs from the weight $\Omega_N$ ($R_c \neq 0$; Eq. (7.10)) in that the former one is a local surface density on $\tilde{\mathcal{R}}_N$ (infinitesimal volume of $\mathcal{Q}_N$ divided by the associated infinitesimal area of $\tilde{\mathcal{R}}_N$) while the latter one is a volume in $\mathcal{Q}_N$ (finite sub-volume of $\mathcal{Q}_N$ associated with a finite area of $\mathcal{R}_N(\mathbf{r})$), i.e. $\Gamma_N$ is not the limit of $\Omega_N$ when $R_c \to 0$. An approximate relationship between the two quantities for small $R_c$ values can be obtained by assuming that: (i) the surface density of random walks is approximately constant over $\mathcal{R}_N(\mathbf{r}(\mathbf{q}))$ in the neighborhood of $\mathbf{q}$; (ii) this surface density is approximately equal to $\Gamma(\mathbf{q})$; (iii) the effect of the curvature of $\mathcal{R}_N(\mathbf{r}(\mathbf{q}))$ in $\mathbb{R}^{3N}$ can be neglected. In this case, $\Omega_N(\mathbf{q}, R_c)$ should be approximately equal (for a given $\mathbf{q}$) to $\Gamma_N(\mathbf{q})$ multiplied by the area of an$n$-dimensional hyperdisc of radius $R_c$:

$$\Omega_N(\mathbf{q}, R_c) \approx \pi^{n/2} R_c^n \gamma^{-1}(n/2 + 1)\, \Gamma_N(\mathbf{q}) \quad \text{for small } R_c \ , \tag{7.17}$$

where $\gamma$ is the Euler gamma function, with $\gamma(n/2 + 1) = 2^{-(n+1)/2}\pi^{1/2} n!!$ for $n$ odd (always the case here). Using Eq. (7.2) for odd $n$ this can be rewritten in terms of the fractional

coverage function $f_N$ (Eq. (7.12)) as:

$$f_N(\mathbf{q}, R_\mathrm{c}) \approx \left[2^{(n-1)/2} n!!\right]^{-1} \Gamma_N(\mathbf{q})\, R_\mathrm{c}^n \quad \text{for small } R_\mathrm{c}\ . \tag{7.18}$$

Note that due to curvature effects one should not expect this equation to be exactly satisfied, even in the limit $R_\mathrm{c} \to 0$ (i.e. in the sense of evaluating $\lim_{R_\mathrm{c}\to 0} R_\mathrm{c}^{-M} f_N(\mathbf{q}, R_\mathrm{c})$).

### 7.2.1 Setup of the numerical experiments

A systematic (grid-based) approach is used to sample the random walk ensemble $\mathcal{W}_N$, for chain lengths ranging from $N = 3$ to 6 beads. This approach involves the regular paving of $\mathcal{Q}_N$ using $G = g^n$ grid cells of volume $G^{-1}V_{\mathcal{Q}_N}$ centered at grid points $\{\mathbf{q}_k | k = 1, \ldots, G\}$, $g$ being the number of cell subdivisions along one dimension (for simplicity, this number is chosen identical for all angle-cosine as well as dihedral angle variables). Grid-based sampling is in principle the most appropriate method when sufficiently large $g$ values are computationally affordable, because it is deterministically reproducible and guarantees a rigorously homogeneous sampling throughout $\mathcal{Q}_N$. Note, however, that when used in combination with too small $g$ values, it may introduce a systematic bias in the sampling (in which case a random sampling approach might be more adequate).

Apart from the number of beads $N$, the bond length $b$ is the only free parameter in the considered random walk ensembles. Because $b$ has the dimension of a length, all monitored properties scale in a predictable manner with $b$. This parameter is thus set to unity in all calculations without affecting the generality of the results.

In order to evaluate the finite-cutoff shape probability density $P_N(\mathbf{q}, R_\mathrm{c})$ at a given grid point $\mathbf{q}_k$, the volume $\Omega_N(\mathbf{q}_k, R_\mathrm{c})$ of Eq. (7.10) is estimated by the number of structures $\mathbf{q}_{k'}$ on the grid satisfying the involved cutoff condition relative to $\mathbf{q}_k$, amplified by $G^{-1}V_{\mathcal{Q}_N}$. These estimates are then used to calculate the corresponding finite-cutoff probability density $P_N(\mathbf{q}_k, R_\mathrm{c})$ via Eq. (7.11), where the integral in the denominator is replaced by a discrete sum over all grid points. The (gridded) fractional coverage function $f_N(\mathbf{q}_k, R_\mathrm{c})$ was evaluated similarly via Eq. (7.12). These two functions are computed for a discrete set of cutoff values $R_\mathrm{c}$ usually corresponding to $R_\mathrm{c}/R_N^{**} = 0.1$, 0.2, 0.4, and 0.6 (for $N = 5$, the last two values are replaced by 0.25 and 0.45; for $N = 6$ the first value is replaced with 0.15 and the last value omitted). The computational cost of the above calculation (one RMSD calculation for each unique pair of distinct structures) is $\mathcal{O}(G(G-1)/2) \sim \mathcal{O}(g^{2n})$, which is only tractable for reasonable values of $g$ along with a rather small number of beads (e.g., $N \leq 6 \to g^{2n} \leq g^{14}$). For this reason, the analysis using finite cutoff distances is not extended beyond 6 beads here.

In order to evaluate the local shape probability density $P_N(\mathbf{q}, 0)$ at a given grid point $\mathbf{q}_k$, the surface density $\Gamma_N(\mathbf{q}_k)$ of Eq. (7.13) was estimated using a finite-difference approach. More precisely, for a grid point $\mathbf{q}_k$ (reference structure), the $\mathbf{q}$ vector is increased or decreased by half the grid spacing along each of the $n$ dimensions, resulting in $2n$ slightly altered structures (shifted structures). The reference and shifted structures are transformed to $\mathcal{C}_N$ and the latter

ones roto-translationally fitted onto the former one. The corresponding $n \times 3N$ Cartesian displacements, divided by the grid spacing and scaled by $N^{1/2}$, provide finite-difference estimates for the elements of the first $n$ rows of the Jacobian matrix . The Jacobian is then completed by the $N + 5$ orthogonal unit vectors. This construction requires $N + 5$ matrix inversions. Finally, the (absolute value of the) inverse of this Jacobian determinant provided the required value for $\Gamma_N(\mathbf{q}_k)$. These estimates are then used to calculate the corresponding (gridded) local probability density $P_N(\mathbf{q}_k, 0)$ via Eq. (7.14), where the integral involved in $I_N(\mathbf{q}_k)$ is replaced by a discrete sum over all grid points. The value of $g$ employed for these calculations is chosen to be even, so that the reference walk (a grid-cell center) can never contain angles equal to 0 or $\pi$. Whenever this situation occurs for a shifted walk, the corresponding angle cosine was simply set to $\pm 0.9999$ instead of $\pm 1$. This avoids the need for a special handling of this situation (the resulting error being essentially negligible). The computational cost of the above calculation (one $\Gamma_N$ calculation for each structure) is $\mathcal{O}(G) = \mathcal{O}(g^n)$, which represents a more favorable scaling compared to the corresponding calculation at finite cutoff (see above), i.e. it remains tractable for reasonable values of $g$ up to a larger number of beads (e.g., $N \leq 9 \rightarrow g^M \leq g^{13}$). For this reason, the analysis using infinitesimal cutoffs was extended up to 9 beads. In order to obtain more precise coordinates for the densest shape $\mathbf{q}_N^{\#}$ (the shapes maximizing $P_N(\mathbf{q}, 0)$), a grid-focusing approach was used, whereby the grid cell containing the best structure at a relatively low $G$ value is iteratively rediscretized by a full grid of $G$ points. This can be done reliably up to 6 beads only.

## 7.2.2 Numerical results

We present both infinitesimal and finite-cutoff results for $N = 3, \ldots, 6$. We first consider the different local shape probabilities. Then we report the derived densest and barycentric shapes.

**Random Walks for N = 3.** The results for $N = 3$ beads are shown in Fig. 7.3. The internal coordinate vector $\mathbf{q}$ consists of the cosine of the single angle $\theta_1$ defined by the three beads. The local shape probability density $P_3(\mathbf{q}, 0)$ is displayed in Fig. 7.3a as a function of $\theta_1$ for three different grid spacings $g$ (10, 100, and 1000). The results for the three $g$ values are consistent, the curves corresponding to $g = 100$ and 1000 being nearly indistinguishable (indicating a sufficient accuracy of the finite-difference approximation to the Jacobian). As expected, the shapes centered at $\theta_1 = 0, \pi$ are characterized by a vanishing probability. The distribution shows a single maximum and is slightly biased toward open angles ($> 90°$). The maximum (densest shape $\mathbf{q}_3^{\#}$) is located at $\theta_1^{\#} = 105.5°$ and associated with a local shape probability density $P_3(\mathbf{q}_3^{\#}, 0) = 1.28$. This indicates that the local neighborhood of this central structure is 28% more populated (in terms of random walk density) than the corresponding average over all possible shapes or, equivalently, that the corresponding shape is 28% more likely than any shape taken at random. This difference may seem modest because it expresses a bias relative to the average over all shapes, including the most likely ones. However, pairwise comparisons can show more dramatic effects. For example, the densest shape is about 15 times more likely compared to the one centered at $\theta_1 = 5°$.

The finite-cutoff shape probability density $P_3(\mathbf{q}, R_c)$ is displayed in Fig. 7.3b as a function of $\theta_1$ for four different cutoff values $R_c$. As expected, the curve corresponding to the lowest

$R_c$ is the closest to the limiting case $R_c \to 0$ (Fig. 7.3a). With increasing $R_c$, the bias in the distribution and the location of the maximum progressively shift in the direction of closed angles ($< 90°$). For the largest $R_c$ value considered here ($R_c = 0.57b > R_3^*$; see below), the maximum no longer corresponds to a single $\theta_1$ value, but to a range thereof. This arises because for such a large cutoff several shapes are able to encompass the entire RW ensemble.

The dependence of $P_3(\mathbf{q}, R_c)$ on the cutoff value is characterized in more detail in Fig. 7.3. The left panel illustrates the maximum $f_3^{max}(R_c)$, mean $f_3^{mean}(R_c)$, and minimum $f_3^{min}(R_c)$ values (over shapes centered at all points of $\mathcal{Q}_3$) of the fractional coverage function $f_3(\mathbf{q}, R_c)$ as a function of $R_c$. The central panel shows the associated maximum $P_3^{max}(R_c)$, mean $P_3^{mean}(R_c)$, and minimum $P_3^{min}(R_c)$ values of the finite-cutoff shape probability density $P_3(\mathbf{q}, R_c)$. The right panel displays the range of $\theta_1$ values associated with the central structures of the most likely shapes, i.e. those corresponding to $f_3^{max}$ and $P_3^{max}$) as well as the average value of $\theta_1$ over this set. The function $f_3^{max}$, i.e. the fraction of the RW ensemble encompassed by the most likely shape for a given $R_c$, increases from zero at $R_c = 0$ (infinitesimal shape) to 1 (the most probable shapes encompass the entire ensemble). This function reaches 1 at a cutoff value $R_3^* = 0.486b$ for a specific shape (barycentric shape $\mathbf{q}_3^*$) characterized by $\theta_1^* = 74.2°$ and $P_3(\mathbf{q}_3^*, R_3^*) = 1.13$. This shape is the one that encompasses the entire ensemble for the smallest possible value of $R_c$. As could be anticipated from Fig. 7.3b, the $\theta_1$ angle associated with the most likely shape decreases upon increasing $R_c$ from 0 to $R_3^*$. Over a sizable range of $R_c$ values ($0 \le R_c \le 0.4b$), this most likely shape is consistently more probable than any shape taken at random. The function $f_3^{min}$, i.e. the fraction of the RW ensemble encompassed by the least probable shape for a given $R_c$, also increases from zero at $R_c = 0$ to 1. This function reaches 1 at a cutoff value $R_3^{**} = 0.943b$. Above this $R_c$ value, all shapes encompass the entire ensemble. As expected, the range of $\theta_1$ values satisfying $f_3(\mathbf{q}, R_c) = 1$ widens upon increasing $R_c$ from $R_3^*$ to $R_3^{**}$, while the average $\theta_1$ value over this set slightly increases over this interval.

As a final note concerning the above results for $N = 3$, it is important to stress that although three beads are always contained in a plane, the present results pertain to random walks in three dimensions. In this sense, the probability distribution $p_3(\mathbf{q})$ in the RW ensemble (Eq. (7.3)) is homogeneous in $\cos\theta_1$ (with $\theta_1$ in the range $[0; \pi]$), so that the corresponding average $\cos\theta_1$ value is 0. This is in qualitative agreement with the observation that the most likely shapes have $\theta_1$ angles close to 90°. In two dimensions, however, the corresponding probability distribution would be homogeneous in $\theta_1$ so that the corresponding average $\theta_1$ value would be 0° (if $\theta_1$ is chosen in the range $]-\pi; \pi]$). The results in terms of shape probability distributions would then look quite different.

**Random Walks for $\mathbf{N = 4}$.** The results for $N = 4$ beads are shown in Fig. 7.4. The internal coordinate vector $\mathbf{q}$ consists of the cosines of the two angles $\theta_1$ and $\theta_2$ along with the single dihedral angle $\omega_1$ defined by the four beads.

The local shape probability density $P_4(\mathbf{q}, 0)$ is displayed in Fig. 7.4a as a function of $\cos\theta_1$, $\cos\theta_2$, and $\omega_1$. As expected, the shapes centered at $\theta_1 = 0, \pi$ or $\theta_2 = 0, \pi$ are characterized by a vanishing probability. In addition, due to the symmetry properties of the RMSD metric , the

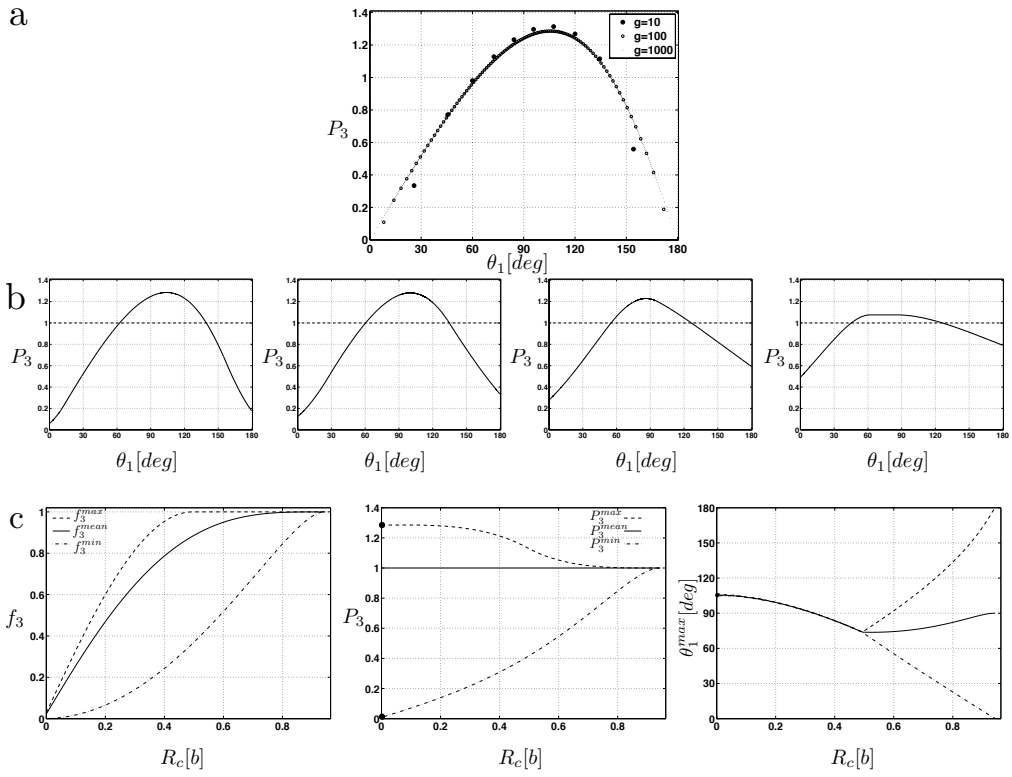



Figure 7.3: (a) Normalized local shape probability distribution $P_3(\mathbf{q}, 0)$ (Eq. (7.14)), where $\mathbf{q} = \{\cos\theta_1\}$, displayed as a function of the single angle $\theta_1$ of the walk. (b) Corresponding normalized finite-cutoff shape probability distribution $P_3(\mathbf{q}, R_c)$ (Eq. (7.11)), displayed as a function of the single angle $\theta_1$ of the walk; left to right: $R_c = 0.09b$, $0.19b$, $0.38b$, and $0.57b$, $b$ being the bond length. (c) Left: maximum ($f_3^{max}$), mean ($f_3^{mean}$), and minimum ($f_3^{min}$) values of the fractional coverage function $f_3(\mathbf{q}, R_c)$ (Eq. (7.12)) over $\mathcal{Q}_3$, displayed as a function of the cutoff $R_c$. Middle: maximum ($P_3^{max}$), mean ($P_3^{mean}$) and minimum ($P_3^{min}$) values of $P_3(\mathbf{q}, R_c)$ over $\mathcal{Q}_3$, displayed as a function of the cutoff $R_c$. Right: maximum, mean and minimum values of the angle $\theta_1$ over the set of structure maximizing $P_3(\mathbf{q}, R_c)$, displayed as a function of the cutoff $R_c$; Dots indicated at $R_c = 0$ in the middle and right panels correspond to the expected values based on the local probability analysis. Note that $P_3$ in (a) and (b) is normalized in terms of $\cos\theta_1$ (not $\theta_1$) so that its average over the graph differs from one. The data in (a) was evaluated using three different numbers of grid points $g = 10$, 100 or 1000. The data in (b) and (c) were evaluated using $g = 10^5$ grid points.

distribution is invariant with respect to the changes $\omega_1 \leftrightarrow -\omega_1$ and $\theta_1 \leftrightarrow \theta_2$. This distribution displays a single maximum and is significantly biased toward open $\theta_1$ and $\theta_2$ angles. The maximum (densest shape $\mathbf{q}_4^{\#}$) is located at $\theta_1^{\#} = \theta_2^{\#} = 137.7°$ and $\omega_1^{\#} = 0.0°$ and associated with a local shape probability density $P_4(\mathbf{q}_4^{\#}, 0) = 1.79$. Note that the presence of a single maximum is not a consequence of the above-mentioned symmetry properties (these merely imply that if the maximum is unique, it must satisfy $\theta_1^{\#} = \theta_2^{\#}$ and $\omega_1^{\#} = 0°$). This specific shape is about three times more likely (in a local sense) than any shape taken at random. Here too, the bias may be much more dramatic when performing pairwise comparisons between shapes. For example, the densest shape is about 35 times more likely than the one centered at $\theta_1 = \theta_2 = 5°$ and $\omega_1 = 0°$.

The finite-cutoff shape probability density $P_4(\mathbf{q}, R_c)$ is displayed in Fig. 7.4 as a function of $\cos\theta_1$, $\cos\theta_2$, and $\omega_1$ for four different cutoff values $R_c$. All graphs preserve the symmetry features described above for $P_4(\mathbf{q}, 0)$. The plot corresponding to the lowest $R_c$ value is again the closest to the limiting case $R_c \to 0$ (Fig. 7.4a). With increasing $R_c$, the bias in the distribution and the location of the maximum progressively shift in the direction of closed angles. For $R_c > 0.4b$, the single maximum splits into two symmetry related (enantiomeric) maxima with opposite $\omega_1 \neq 0$ values. For the largest $R_c$ value considered ($R_c = 0.74b > R_4^*$; see below), these two maxima no longer correspond to single points, but to two regions of the space.

The dependence of $P_4(\mathbf{q}, R_c)$ on the cutoff value is characterized in more details in Fig. 7.4c, analogously to Fig. 7.3c for $N = 3$ (see explanations above). These curves display the same qualitative features as for $N = 3$, although the numerical precision is considerably lower (especially for low $R_c$ values) due to the more limited grid resolution. The fractional coverage function $f_4^{max}$ reaches 1 at a cutoff value $R_4^* = 0.712b$ for a specific (symmetry duplicated, i.e. two enantiomers) shape (barycentric shape $\mathbf{q}_4^*$) characterized by $\theta_1^* = \theta_2^* = 36.0°$ and $\omega_1^* = \pm 162.4°$, and $P_4(\mathbf{q}_4^*, R_4^*) = 1.11$. As could be anticipated from Fig. 7.4b, the $\theta_1$ and $\theta_2$ values associated with the most likely shape (which remain identical to each other) decrease upon increasing $R_c$ from 0 to $R_4^*$, while the corresponding single $\omega_1$ value of 0° splits into two opposite (and increasingly larger) values for $R_c > 0.4b$. Over a sizable range of $R_c$ values ($0 \leq R_c \leq 0.4b$), this most likely shape is consistently more probable than any shape taken at random. The function $f_4^{min}$ reaches 1 at a cutoff value $R_4^{**} = 1.236$. Above this $R_c$ value, all shapes encompass the entire ensemble. As expected, the ranges of $\theta_1$, $\theta_2$, and $\omega_1$ values satisfying $f_4(\mathbf{q}, R_c) = 1$ widen upon increasing $R_c$ from $R_4^*$ to $R_4^{**}$.

**Random Walks for N = 5.** The results for $N = 5$ beads are displayed in Fig. 7.5. The internal coordinate vector $\mathbf{q}$ consists of the cosines of the three angles $\theta_1$, $\theta_2$, and $\theta_3$ along with the two dihedral angles $\omega_1$ and $\omega_2$ defined by the five beads.

The local shape probability density $P_5(\mathbf{q}, 0)$ is shown in Fig. 7.5a in the form of a maximum value (over all possible $\theta_1$, $\theta_2$ and $\theta_3$ combinations) as a function of the $\omega_1$ and $\omega_2$ dihedral angles. It is verified that the full (five-dimensional) distribution (not shown) satisfies the expected symmetry properties. These translate at the level of the two-dimensional maximum-

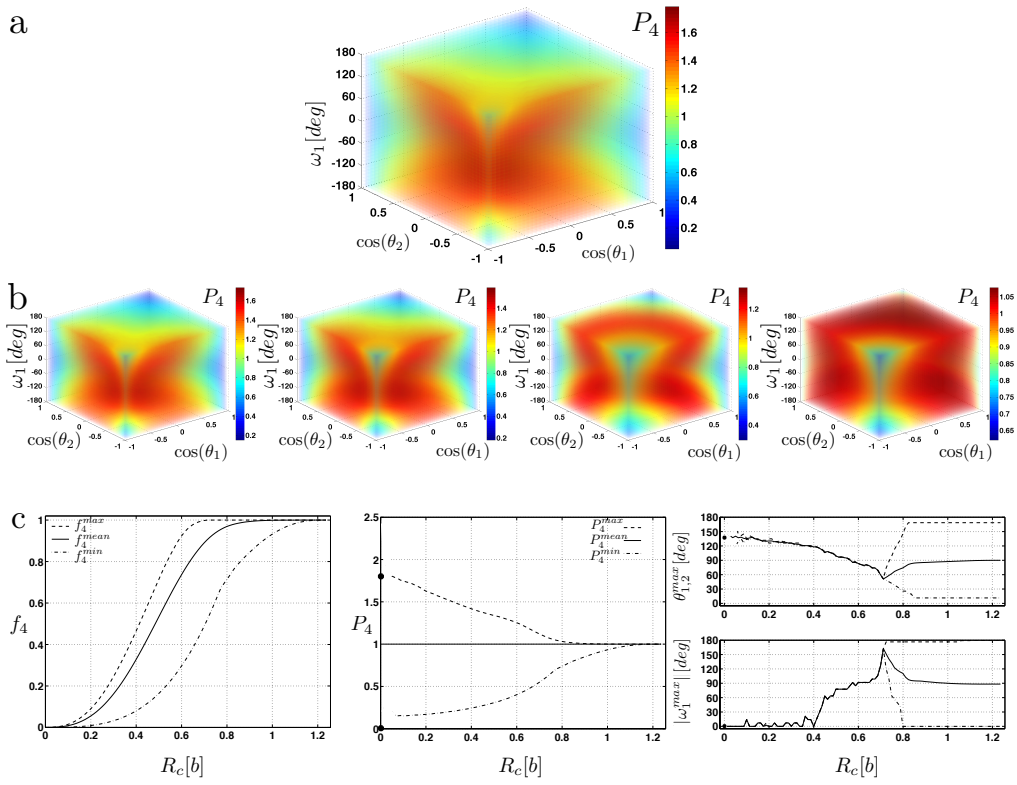



Figure 7.4: (a) Normalized local shape probability distribution $P_4(\mathbf{q}, 0)$ (Eq. (7.14)), where $\mathbf{q} = \{\cos\theta_1, \cos\theta_2, \omega_1\}$, displayed as a function of the angle cosines $\cos\theta_1$ and $\cos\theta_2$ and the single dihedral angle $\omega_1$ of the walk. (b) Corresponding normalized finite-cutoff shape probability distribution $P_4(\mathbf{q}, R_c)$ (Eq. (7.11)), displayed as a function of the angle cosines $\cos\theta_1$ and $\cos\theta_2$ and the single dihedral angle $\omega_1$ of the walk; left to right: $R_c = 0.12b$, $0.25b$, $0.50b$, and $0.74b$, $b$ being the bond length. (c) Left: maximum ($f_4^{max}$), mean ($f_4^{mean}$) and minimum ($f_4^{min}$) values of the fractional coverage function $f_4(\mathbf{q}, R_c)$ (Eq. (7.12)) over $\mathcal{Q}_4$, displayed as a function of the cutoff $R_c$. Middle: maximum ($P_4^{max}$), mean ($P_4^{mean}$), and minimum ($P_4^{min}$) values of $P_4(\mathbf{q}, R_c)$ over $\mathcal{Q}_4$, displayed as a function of the cutoff $R_c$. Right: maximum, mean, and minimum values of the angles $\theta_1$ and $\theta_2$ as well as of the dihedral angle $\omega_1$ (absolute value) over the set of structure maximizing $P_4(\mathbf{q}, R_c)$, displayed as a function of the cutoff $R_c$. Dots indicated at $R_c = 0$ in the middle and right panels correspond to expected values based on the local probability analysis. The data in (a) were evaluated using $g = 200$ grid points per dimension, and the data in (b) and (c) using $g = 51$ grid points per dimension.

value projection to invariances with respect to the changes $\omega_1 \leftrightarrow \omega_2$ and $\omega_1, \omega_2 \leftrightarrow -\omega_1, -\omega_2$. The distribution displays a single maximum and is significantly biased toward flat *cis-cis* structures. The maximum (densest shape $\mathbf{q}_5^{\#}$) is located at $\theta_1^{\#} = \theta_3^{\#} = 154.5°$, $\theta_2^{\#} = 143.7°$, and $\omega_1^{\#} = \omega_2^{\#} = 0.0°$ and associated with a local shape probability density $P_5(\mathbf{q}_5^{\#}, 0) = 2.94$. Here too, the presence of a single maximum is not a consequence of the above-mentioned symmetry properties (these merely imply that if the maximum is unique, it must satisfy $\theta_1^{\#} = \theta_3^{\#}$ and $\omega_1^{\#} = \omega_2^{\#} = 0°$). This specific shape is about three times more likely (in a local sense) than any shape taken at random.

The finite-cutoff shape probability density $P_5(\mathbf{q}, R_c)$ is shown in Fig. 7.5b, also in the form of a maximal-value projection, for four different cutoff values. All graphs preserve the symmetry features described above for $P_5(\mathbf{q}, 0)$. The curve corresponding to the lowest $R_c$ value is again the closest to the limiting case $R_c \to 0$ (Fig. 7.5a). With increasing $R_c$, the bias in the distribution progressively shifts from flat *cis-cis* structures in the direction of *gauche*$^{\pm}$-*gauche*$^{\mp}$ structures, while the single maximum splits into four symmetry-related (enantiomeric forms and reverse bead order) maxima with $\omega_1, \omega_2 \neq 0$ values.

The dependence of $P_5(\mathbf{q}, R_c)$ on the cutoff value is characterized in more details in Fig. 7.5c, analogously to Fig. 7.3c and Fig. 7.4c for $N = 3, 4$ (see explanations above). These curves display the same qualitative features as for $N = 3$ and 4, although the numerical precision is again considerably lower (especially for low $R_c$ values). The fractional coverage function $f_5^{max}$ reaches 1 at a cutoff value $R_5^* = 0.873b$ for a specific shape (barycentric shape $\mathbf{q}_5^*$) that is four-fold replicated by symmetry and associated with $P_5(\mathbf{q}_5^*, R_5^*) = 1.07$. One of these structures corresponds to $\theta_1^* = \theta_3^* = 81°$, $\theta_2^* = 67°$, $\omega_1^* = 166°$, and $\omega_2^* = -83°$. The other three are obtained by the changes $\theta_1^*, \omega_1^* \leftrightarrow \theta_3^*, \omega_2^*$ and/or $\omega_1^*, \omega_2^* \leftrightarrow -\omega_1^*, -\omega_2^*$. As could be anticipated from Fig. 7.5b, the dihedral angles associated with the most likely shape tend to shift away from flat *cis-cis* upon increasing $R_c$ from 0 to $R_5^*$. Simultaneously, the angles tend to shift toward lower values. The function $f_5^{min}$ reaches 1 at a cutoff value $R_5^{**} = 1.600$. Above this $R_c$ value all shapes encompass the entire ensemble.

**Random Walks for N = 6.** The results for $N = 6$ beads are displayed in Fig. 7.6. The internal coordinate vector $\mathbf{q}$ consists of the cosines of the four angles $\theta_1$, $\theta_2$, $\theta_3$ and $\theta_4$ along with the three dihedral angles $\omega_1$, $\omega_2$ and $\omega_3$ defined by the six beads.

The local shape probability density $P_6(\mathbf{q}, 0)$ is displayed in Fig. 7.6a in the form of the maximal value (over all possible $\theta_1$, $\theta_2$, $\theta_3$, and $\theta_4$ combinations) as a function of the $\omega_1$, $\omega_2$, and $\omega_3$ dihedral angles. It is verified that the full (seven-dimensional) distribution (not shown) satisfies the expected symmetry properties. These translate at the level of the three-dimensional maximum-value projection to invariances with respect to the changes $\omega_1 \leftrightarrow \omega_3$ and $\omega_1, \omega_2, \omega_3 \leftrightarrow -\omega_1, -\omega_2, -\omega_3$. The distribution shows a single maximum and is significantly biased toward flat *cis-cis-cis* structures. The maximum (densest shape $\mathbf{q}_6^{\#}$) is located at $\theta_1^{\#} = \theta_4^{\#} = 163.5°$, $\theta_2^{\#} = \theta_3^{\#} = 151.8°$, and $\omega_1^{\#} = \omega_2^{\#} = \omega_3^{\#} = 0.0°$ and associated with a local shape probability density $P_6(\mathbf{q}_6^{\#}, 0) = 10.05$. Here too, the presence of a single maximum is not a consequence of the above-mentioned symmetry properties (these merely imply that if

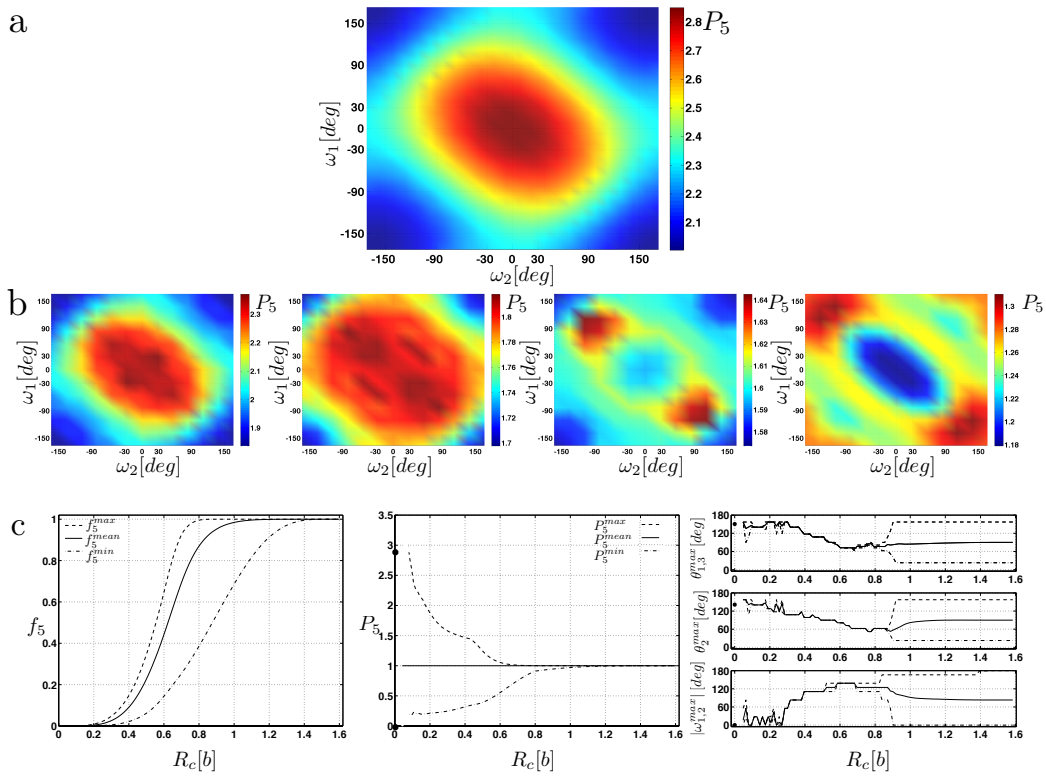



Figure 7.5: (a) Normalized local shape probability distribution $P_5(\mathbf{q}, 0)$ (Eq. (7.14)), where $\mathbf{q} = \{\cos\theta_1, \cos\theta_2, \cos\theta_3, \omega_1, \omega_2\}$, displayed as a maximum projection onto the two dihedral angles $\omega_1$ and $\omega_2$ of the walk. (b) Corresponding normalized finite-cutoff shape probability distribution $P_5(\mathbf{q}, R_c)$ (Eq. (7.11)), displayed as a maximum projection onto the two dihedral angles $\omega_1$ and $\omega_2$ of the walk; left to right: $R_c = 0.16b$, $0.32b$, $0.40b$ and $0.72b$, $b$ being the bond length. (c) Left: maximum ($f_5^{max}$), mean ($f_5^{mean}$), and minimum ($f_5^{min}$) values of the fractional coverage function $f_5(\mathbf{q}, R_c)$ (Eq. (7.12)) over $\mathcal{Q}_5$, displayed as a function of the cutoff $R_c$. Middle: maximum ($P_5^{min}$), mean ($P_5^{mean}$), and minimum ($P_5^{max}$) values of $P_5(\mathbf{q}, R_c)$ over $\mathcal{Q}_5$, displayed as a function of the cutoff $R_c$. Right: maximum, mean and minimum values of the outer angles $\theta_1$ and $\theta_3$, of the central angle $\theta_2$, and of the two dihedral angles $\omega_1$ and $\omega_2$ (absolute values) over the set of structure maximizing $P_5(\mathbf{q}, R_c)$, displayed as a function of the cutoff $R_c$. Dots indicated at $R_c = 0$ in the middle and right panels correspond to expected values based on the local probability analysis. The data in (a) was evaluated using $g = 24$ grid points per dimension, and the data in (b) and (c) using $g = 13$ grid points per dimension.

the maximum is unique, it must satisfy $\theta_1^{\#} = \theta_4^{\#}$, $\theta_2^{\#} = \theta_3^{\#}$ and $\omega_1^{\#} = \omega_2^{\#} = \omega_3^{\#} = 0°$). This specific shape is about ten times more likely (in a local sense) than any shape taken at random.

The finite-cutoff shape probability density $P_6(\mathbf{q}, R_c)$ is displayed in Fig. 7.6b, also in the form of a maximum-value projection, for three different cutoff values $R_c$. All graphs preserve the symmetry features described above for $P_6(\mathbf{q}, 0)$. The curve corresponding to the lowest $R_c$ value is again the closest to the limiting case $R_c \to 0$ (Fig. 7.6). With increasing $R_c$, the bias in the distribution progressively shifts from flat *cis-cis-cis* structures in the direction of *gauche*$^{\pm}$-*gauche*$^{+}$-*gauche*$^{\mp}$ or *gauche*$^{\pm}$-*gauche*$^{-}$-*gauche*$^{\mp}$ structures, while the single maximum splits into a pair (two alternative $\omega_2$ values) of two symmetry-related (enantiomeric) maxima with $\omega_1, \omega_2, \omega_3 \neq 0$ values.

The dependence of $P_6(\mathbf{q}, R_c)$ on the cutoff value is characterized in more details in Fig. 7.6c, analogously way to Fig. 7.3c, Fig. 7.4c and Fig. 7.5c for $N = 3, 4, 5$ (see explanations above). These curves display the same qualitative features as for $N = 3, 4$, and 5, although the numerical precision is again considerably lower (especially for low $R_c$ values). The function $f_6^{max}$ reaches 1 at a cutoff value $R_6^* = 0.998b$ for a specific pair of shapes (barycentric shapes $\mathbf{q}_6^*$) that are four-fold replicated by symmetry and associated with $P_6(\mathbf{q}_6^*, R_6^*) = 1.06$. One pair of these structures corresponds to $\theta_1^* = \theta_4^* = 66°$, $\theta_2^* = \theta_3^* = 78°$, $\omega_1^* = 144°$, $\omega_2^* = \pm 72°$, and $\omega_3^* = -144°$. The other pairs are obtained by the changes $\omega_1^* \leftrightarrow \omega_3^*$ or/and $\omega_1^*, \omega_2^*, \omega_3^* \leftrightarrow -\omega_1^*, -\omega_2^*, -\omega_3^*$. The evolution of the angles and dihedral angles associated with the most likely shapes upon increasing $R_c$ from 0 to $R_6^*$ is difficult to assess in detail (numerical noise for small $R_c$), but agrees with the trends observed in Fig. 7.6b. The function $f_6^{min}$ reaches 1 at a cutoff value $R_6^{**} = 1.899$. Above this $R_c$ value all shapes encompass the entire ensemble.

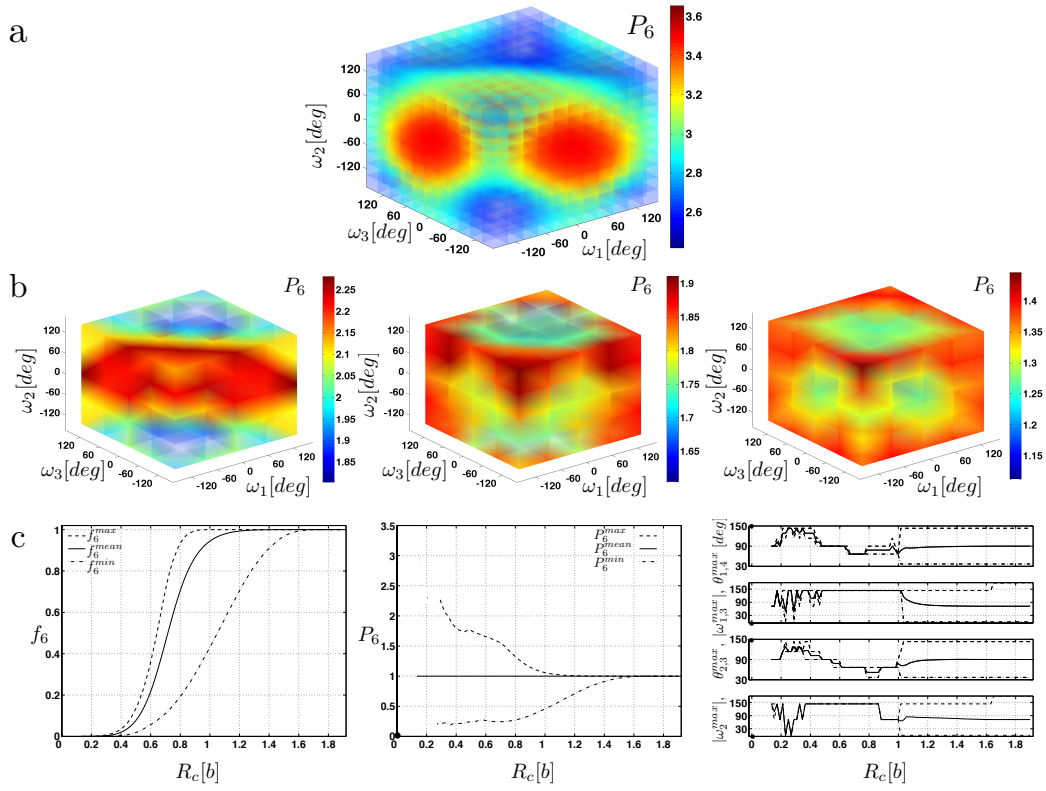



Figure 7.6: (a) Normalized local shape probability distribution $P_6(\mathbf{q}, 0)$ (Eq. (7.14)), where $\mathbf{q} = \{\cos\theta_1, \cos\theta_2, \cos\theta_3, \cos\theta_4, \omega_1, \omega_2, \omega_3\}$, displayed as a maximum projection onto the three dihedral angles $\omega_1$, $\omega_2$ and $\omega_3$ of the walk. (b) Corresponding normalized finite-cutoff shape probability distribution $P_6(\mathbf{q}, R_c)$ (Eq. (7.11)), displayed as a maximum projection onto the three dihedral angles $\omega_1$, $\omega_2$ and $\omega_3$ of the walk; left to right: $R_c = 0.29b$, $0.38b$ and $0.76b$, $b$ being the bond length. (c) Left: maximum ($f_6^{max}$), mean ($f_6^{mean}$), and minimum ($f_6^{min}$) values of the fractional coverage function $f_6(\mathbf{q}, R_c)$ (Eq. (7.12)) over $\mathcal{Q}_6$, displayed as a function of the cutoff $R_c$. Middle: maximum ($P_6^{max}$), mean ($P_6^{mean}$), and minimum ($P_6^{min}$) values of $P_6(\mathbf{q}, R_c)$ over $\mathcal{Q}_6$, displayed as a function of the cutoff $R_c$. Right: maximum, mean, and minimum values of the outer angles $\theta_1$ and $\theta_4$, of the central angles $\theta_2$ and $\theta_3$, of the two outer dihedral angles $\omega_1$ and $\omega_3$ (absolute value) and of the central dihedral angle $\omega_2$ (absolute value) over the set of structure maximizing $P_6(\mathbf{q}, R_c)$, displayed as a function of the cutoff $R_c$. Dots indicated at $R_c = 0$ in the middle and right panels correspond to expected values based on the local probability analysis. The data in (a) were evaluated using $g = 12$ grid points per dimension, and the data in (b) and (c) using $g = 5$ grid points per dimension.

**The densest and the barycentric shapes.** The identified densest and the barycentric shapes are shown in Fig. 7.7. The corresponding structural parameters are given in Table 7.1.

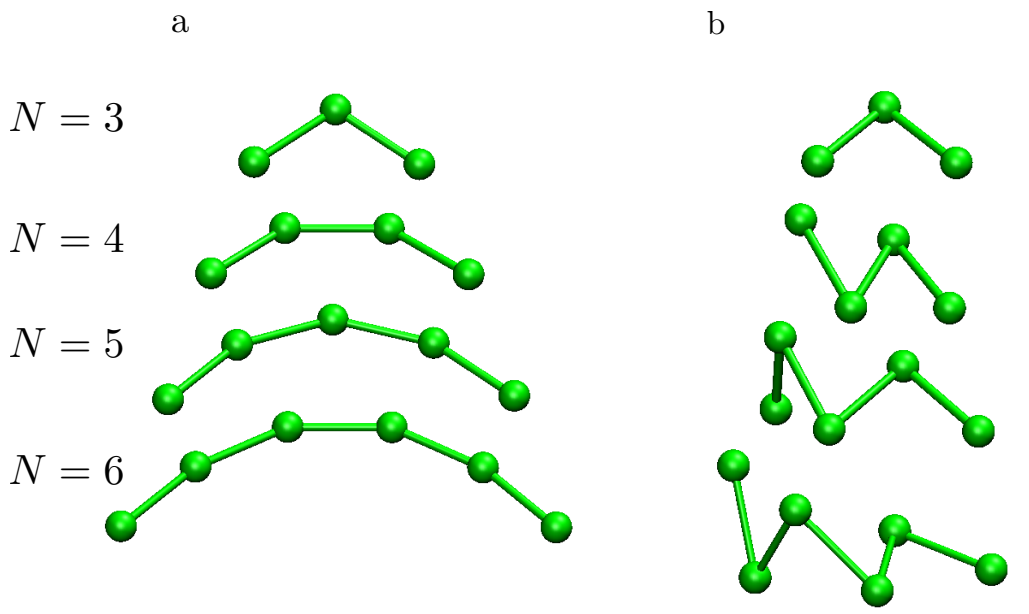



Figure 7.7: (a) Central structures associated with the densest shapes. (b) Central structures associated with the barycentric shape. The different structures come from to the RW ensembles $\mathcal{W}_N$ for different numbers of beads $N = 3, \ldots, 6$. All structures are drawn according to the internal coordinates $\mathbf{q}^{\#}$ and $\mathbf{q}^{*}$ reported in Table 7.1. For $N = 4, \ldots, 6$, only one of the 2 or 4 alternative barycentric shapes is represented ($N = 4$: $\omega_1 = 162.4°$; $N = 5$: $\omega_1 = 166°$, $\omega_2 = -83°$; $N = 6$: $\omega_1 = 144°$, $\omega_2 = 72°$, $\omega_1 = -144°$). The parameters of $\mathbf{q}^{\#}$ for $N = 3, \ldots, 6$ are refined to a precision of at least $1°$. The precisions of the parameters of $\mathbf{q}^{*}$ is limited by the number of grid points $g$ used (see caption of Table 7.1 for details).

The central structures associated with the densest shapes are all planar up to $N = 6$, with bond angles progressively opening with increasing $N$ (Fig. 7.7 a). In contrast, the central structures associated with the barycentric shapes are not planar (except for $N = 3$), and do not present obvious systematic trends upon increasing $N$ (Fig. 7.7). This may be due to the low precision in the determination of these structures for $N = 5$ and 6 ($g = 13$ and $g = 5$, respectively). Based on the present results, it is not possible to determine whether the above properties of the densest and barycentric structures also hold for larger $N$ values.

The threshold cutoff-radii $R_N^*$ and $R_N^{**}$ systematically increase with increasing $N$. This is expected since the size of the accessible conformational space also increases. $R_N^*$ values can be optimally fitted (for $N = 3, \ldots, 6$) by the logarithmic expression $R_N^* \approx (0.7393 \log(N) - 0.3207)\, b$. The study of $R_N^{**}$, the diameter of the conformation space, is the topic of Section 7.3.

| infinitesimal cutoff | | | | | |
|---|---|---|---|---|---|
| $N$ | $\{\theta^{\#}\}$ [deg] | $\{\omega^{\#}\}$ [deg] | $P_N(\mathbf{q}_N^{\#}, 0)$ | | |
| 3 | 105.5 | - | 1.28 (1.15) | | |
| 4 | 137.7/137.7 | 0.0 | 1.79 (1.50) | | |
| 5 | 154.5/143.7/154.5 | 0.0/0.0 | 2.94 (2.34) | | |
| 6 | 163.5/151.8/151.8/163.5 | 0.0/0.0/0.0 | 10.05 (7.60) | | |
| **finite cutoff** | | | | | |
| $N$ | $\{\theta^{*}\}$ [deg] | $\{\omega^{*}\}$ [deg] | $P_N(\mathbf{q}_N^{*}, R_N^{*})$ | $R_N^{*}[b]$ | $R_N^{**}[b]$ |
| 3 | 74.2 | - | 1.13 | 0.486 | 0.943 |
| 4[a] | 36.0/36.0 | $\pm$ 162.4 | 1.11 | 0.712 | 1.236 |
| 5[b] | 81/67/81 | $\pm$ 166/$\mp$ 83, $\mp$ 83/$\pm$ 166 | 1.07 | 0.873 | 1.600 |
| 6[c] | 66/78/78/66 | $\pm$ 144/(+72,-72)/$\mp$ 144 | 1.06 | 0.998 | 1.899 |

Table 7.1: Parameters characterizing the local shape probability density $P_N(\mathbf{q}, 0)$ (top) and the finite-cutoff shape probability density $P_N(\mathbf{q}, R_c)$ (bottom) for the densest and barycentric shapes of the RW ensembles $\mathcal{W}_N$ with different numbers of beads $N = 3, \ldots, 6$. The parameters are: the central structure $\mathbf{q}_N^{\#}$ of the densest shape (internal coordinates $\{\theta^{\#}\}$ and $\{\omega^{\#}\}$), the local probability density $P_N(\mathbf{q}_N^{\#}, 0)$ associated with this shape (Eq. (7.14)), the central structure $\mathbf{q}_N^{*}$ of the barycentric shape (internal coordinates $\{\theta^{*}\}$ and $\{\omega^{*}\}$), the smallest cutoff radius $R_N^{*}$ for which this shape encompasses the entire ensemble, the finite-cutoff probability density $P_N(\mathbf{q}_N^{*}, R_N^{*})$ associated with these shape and cutoff (Eq. (7.11)), and the cutoff radius $R_N^{**}$ above which all shapes encompass the entire ensemble (landscape diameter). The data were evaluated using grid-based sampling in the internal-coordinate space. The number of grid points per dimension was set to $g = 10^4$, 200, 24, and 12 (local density) or $g = 10^5$, 51, 13, and 5 (finite-cutoff density) for $N = 3, \ldots, 6$. The parameters of $\mathbf{q}_N^{\#}$ for $N = 3, \ldots, 6$ were further refined to a precision of at least $1^\circ$ using grid focusing. Table footnotes: (a) two symmetry-related barycentric shapes (enantiomers $\omega_1^{*} \leftrightarrow -\omega_1^{*}$); (b) precisions of reported $\theta^{*}$ and $\omega^{*}$ parameters are only about 20 and $30^\circ$, respectively; four symmetry-related barycentric shapes (enantiomers $\omega_1^{*}, \omega_2^{*} \leftrightarrow -\omega_1^{*}, -\omega_2^{*}$ and bead-order inversion $\omega_1^{*} \leftrightarrow \omega_2^{*}$); (c) precisions of reported $\theta^{*}$ and $\omega^{*}$ parameters are only about 30 and $70^\circ$, respectively; two alternative $\omega_2^{*}$ values, each with four symmetry-related barycentric shapes (enantiomers $\omega_1^{*}, \omega_2^{*}, \omega_3^{*} \leftrightarrow -\omega_1^{*}, -\omega_2^{*}, -\omega_3^{*}$, and bead-order inversion $\omega_1^{*} \leftrightarrow \omega_3^{*}$).

## 7.3 The landscape diameter of linear chains

So far we have analyzed the implications of the RMSD metric on the neighborhood relation between linear chains. Both densest and barycentric structures in the RW ensemble have been reported. We have also observed that, after a certain RMSD cutoff value $R_N^{**}$, all structures were neighboring all other structures. In this section, we use a combination of black-box optimization and analytical geometry to derive an $N$-dependent upper bound for this cutoff value: the conformational landscape diameter.

### 7.3.1 Preliminaries

Compared to the previous section we use slightly adapted notations and definitions that are outlined below.

**Representations, RMSD, and related quantities:** We represent two arbitrary configurations of $N$ points (or beads) by the matrices $X$, $Y \in \mathbb{R}^{3\times N}$. Each column in $X$, $Y$ is denoted by $\mathbf{x}^{(i)}, \mathbf{y}^{(i)}$ and represents the three-dimensional Cartesian coordinates of the $i^{\text{th}}$ bead in the chain. Consecutive beads of mass $m$ are connected by links of length $b$. The minimum RMSD

between $X$ and $Y$ is denoted by $D(X,Y)$ and comprises two steps: (i) Translation of the centers of mass $\mathbf{x}_{\text{cm}}$ and $\mathbf{y}_{\text{cm}}$ of both configurations to the origin, leading to repositioned structures $X_0$ and $Y_0$ with columns $\mathbf{x}_0^{(i)}, \mathbf{y}_0^{(i)}$. (ii) Determination of the optimal rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times 3}$ such that:

$$D^2(X,Y) \doteq \min_{\mathbf{R}} \frac{1}{N} \|\mathbf{R}X_0 - Y_0\|^2 = \min_{\mathbf{R}} \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{R}\mathbf{x}_0^{(i)} - \mathbf{y}_0^{(i)}\|^2 \,. \tag{7.19}$$

The optimal rotation matrix $\mathbf{R}$ is determined using quaternions (Kneller, 1991). $D^2(X,Y)$ can also be expressed in terms of the radii of gyration of $X$ and $Y$, $R_G(X)$ and $R_G(Y)$, as (McLachlan, 1972, 1984):

$$D^2(X,Y) = R_G^2(X) + R_G^2(Y) - 2\frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_0'^{(i)} \cdot \mathbf{y}_0^{(i)} \tag{7.20}$$

with $\mathbf{x}_0'^{(i)} = \mathbf{R}\mathbf{x}_0^{(i)}$. The term $\frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_0^{'(i)} \cdot \mathbf{y}_0^{(i)}$ describes the structural correlation between $X$ and $Y$ after optimal superposition. It can be re-written as (Betancourt and Skolnick, 2001):

$$\frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_0'^{(i)} \cdot \mathbf{y}_0^{(i)} = \frac{\sum_{i=1}^{N} \mathbf{x}_0'^{(i)} \cdot \mathbf{y}_0^{(i)}}{\sqrt{\sum_{i=1}^{N} \mathbf{x}_0^{(i)2} \sum_{i=1}^{N} \mathbf{y}_0^{(i)2}}} R_G(X) R_G(Y) \,. \tag{7.21}$$

Betancourt and Skolnick (Betancourt and Skolnick, 2001) refer to the fraction in Eq. (7.21) as the aligned correlation coefficient $ACC(X,Y)$. The radius of gyration $R_G$ of a chain $X$ is roto-translation invariant and defined as:

$$R_G^2(X) \doteq \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{x}^{(i)} - \mathbf{x}_{\text{cm}}\|^2 = -\mathbf{x}_{\text{cm}} \cdot \mathbf{x}_{\text{cm}} + \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{x}^{(i)}\|^2 \,. \tag{7.22}$$

From Eq. (7.20), McLachlan derives for two given structures $X$ and $Y$ a relative lower and an upper bound for $D^2(X,Y)$ (McLachlan, 1979, 1984):

$$0 \leq D^2(X,Y) \leq R_G^2(X) + R_G^2(Y) \,. \tag{7.23}$$

**Notation for Random Walks.** We consider again "anchored" Random Walks. A specific RW chain $X$ is generated based on internal coordinates. We again denote by $\theta_i$ the angle between three consecutive beads $\mathbf{x}^{(i)}$, $\mathbf{x}^{(i+1)}$, $\mathbf{x}^{(i+2)}$. The dihedral angle between the two consecutive planes spanned by $\mathbf{x}^{(i)}$, $\mathbf{x}^{(i+1)}$, $\mathbf{x}^{(i+2)}$ and $\mathbf{x}^{(i+1)}$, $\mathbf{x}^{(i+2)}$, $\mathbf{x}^{(i+3)}$ is denoted by $\omega_i$. The internal coordinate vector associated with $X$ then is $\boldsymbol{q}_X \doteq \{\theta_i | i = 1 \ldots N-2, \omega_i | i = 1 \ldots N-3\}$. The direction of each link is chosen uniformly random on a sphere by drawing the $\cos(\theta_i)$ uniformly from $[0,1]$ and the $\omega_i$ uniformly from $[0,\pi]$ (Cantor and Schimmel, 1980). The link length between consecutive beads is set to $b = 1$ if not stated otherwise, the mass of each bead is $m = \frac{1}{N}$. Rigid-body translation and rotation are again removed by placing $\mathbf{x}^{(1)}$ at the origin, $\mathbf{x}^{(2)}$ along the $x$-axis at $(b,0,0)^T$, and embedding the link between $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$ in the $xy$-plane.

### 7.3.2 The maximum RMSD problem for Random Walks

The *Random Walk maximum RMSD problem* (RW-MAX-RMSD) can be stated as a continuous max-min optimization problem. We want to determine the specific pair of RW configurations $(X^N_{\max}, Y^N_{\max})$ with $N$ beads that maximizes the minimal squared RMSD $D^2(X,Y)$ over all possible pairs of $X$ and $Y$:

$$(X^N_{\max}, Y^N_{\max}) = \arg\max_{\mathrm{X,Y}} D^2(X,Y) = \arg\max_{\mathrm{X,Y}} \min_{\mathbf{R}} \frac{1}{N}\|\mathbf{R}X_0 - Y_0\|^2 \,. \tag{7.24}$$

We refer to the pair $(X^N_{\max}, Y^N_{\max})$ as the *most dissimilar* or *extremal shapes* in a RW ensemble in the RMSD sense. The goal is to find a closed-form expression for an upper bound on $D^2(X,Y)$. Although there might exist analytical tools and techniques to prove such an upper bound, we address the problem using black-box optimization techniques. While this approach cannot provide a proof of the upper bound, it can give a very educated guess about the possible extremal shapes. These configurations then serve as a starting point for a rigorous mathematical analysis. The minimal RMSD between this pair is denoted $D_{\max}(N)$ and given by:

$$D_{\max}(N) = \sqrt{D^2(X^N_{\max}, Y^N_{\max})} \,. \tag{7.25}$$

In Section 7.3.1 we showed that the internal minimization problem can be solved analytically by constructing the optimal rotation matrix $\mathbf{R}$ from SVD (Kabsch, 1976, 1978) or by using quaternions (Kneller, 1991, 2005). The distances constraints on the positions of consecutive beads $d_{\mathrm{E}}(\mathbf{p}_i, \mathbf{p}_{i+1}) = b, \quad i = 1, \ldots, N-1$, in the RW chain can be fulfilled by representing the RW's in internal coordinates $\boldsymbol{q}$. A pair of anchored Random Walk chains ($X$,$Y$) of $N$ beads each is represented by $\boldsymbol{q}_{\mathrm{S}} = (\boldsymbol{q}_{\mathrm{X}}, \boldsymbol{q}_{\mathrm{Y}})$. The transformation from internal coordinates to the three-dimensional pair of Cartesian configurations is denoted by $J(\boldsymbol{q}_{\mathrm{S}}) = (X,Y)$.

The outer maximization problem can be formulated as a constrained black-box optimization problem in $n = 2 \cdot (2N-5) = 4N-10$ dimensions. For convenience, we consider the unit hypercube as landscape domain, i.e., candidate solution vectors $\hat{\boldsymbol{q}}_{\mathrm{S}}$ are sampled in $[0,1]^n$. The function $T : \hat{\boldsymbol{q}}_{\mathrm{S}} \in [0,1]^n \rightarrow \boldsymbol{q}_{\mathrm{S}} \in ([0,1]^{2(N-2)}, [0,\pi]^{2(N-3)})$ transforms any unit vector into true angle cosines and dihedrals of the internal RW representation. The black-box objective function $f$ to be maximized for RW-MAX-RMSD then reads:

$$f(\hat{\boldsymbol{q}}_{\mathrm{S}}) \equiv D^2\left(J(T(\hat{\boldsymbol{q}}_S))\right) = D^2(X,Y) = \min_{\mathbf{R}} \frac{1}{N}\|\mathbf{R}X_0 - Y_0\|^2 \,. \tag{7.26}$$

Note that this problem formulation is known *a priori* to become twofold degenerate if two consecutive links in a trial configuration become co-linear. First, the corresponding dihedral angles are then undefined, i.e., the configuration remains the same regardless of their values. Second, the optimal rotation matrix has only rank 1, permitting infinitely many rotations that minimize $D^2(X,Y)$ (McLachlan, 1979; Kneller, 1991). This degeneracy leads to plateau regions in the landscape.

### 7.3.3 Numerical solutions of RW-MAX-RMSD

We numerically explore the RW-MAX-RMSD constrained black-box landscape for pairs of shapes with $N = 3, \ldots, 15$ beads and link length $b = 1$. The dimensionality of the problem is

thus ranging from $n = 2(2N - 5) = 2, \ldots, 50$. The following search algorithms are considered: (i) *Pure Random Search* (PRS), (ii) *Sequential Quadratic Programming* (SQP), and (iii) BLR-CMA-ES. PRS, that is uniform random sampling, is used to check how complex the problem is and as a reference for more advanced search heuristics. For SQP, MATLAB's built-in standard optimizer *fmincon* is used. In the box-constrained black-box optimization setting, *fmincon* uses an active-set sequential quadratic programming scheme with approximate BFGS and line search. BLR-CMA-ES has been described in Section 4.2.2. For all methods, standard parameter settings and MAX_FES $= 10^5 n$ are used. PRS and BLR-CMA-ES use pseudo-random sampling. Each experiment is repeated 25 times. Figure 7.8 summarizes the best found solutions for all methods. We denote the identified maxima by $D_{\max}^{\mathrm{PRS}}$, $D_{\max}^{\mathrm{SQP}}$, and $D_{\max}^{\mathrm{CMA}}$. The

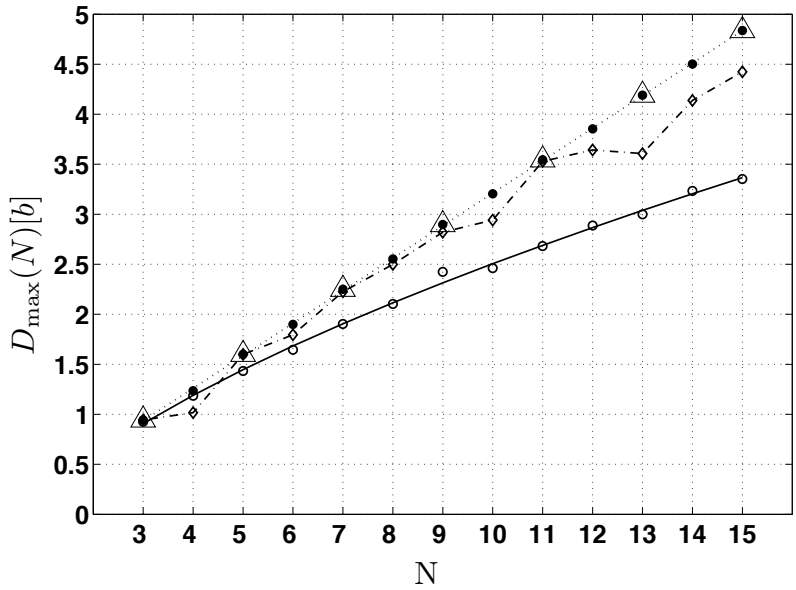



Figure 7.8: Numerical solutions of RW-MAX-RMSD found by PRS (○), SQP (◇), and BLR-CMA-ES (●). The triangles (△) show the conjectured analytical results $D_{\max}(N)$ for odd $N$ (see Section 7.3.4). The solid line shows the best-fit power law to the PRS results, the dotted line shows the best linear fit of the CMA-ES results.

solutions of PRS suggest a power-law relationship between $N$ and $D_{\max}^{\mathrm{PRS}}$ (Fig. 7.8; the best fit is given by $D_{\max}^{\mathrm{PRS}} = 0.7613 N^{0.6074} - 0.5775$). Visual inspection of the extremal shapes does not reveal any obvious geometrical pattern. A general observation, however, is that all maximizing pairs consist of one collapsed and one extended structure. SQP finds better solutions than PRS that do not show a regular scaling. For $N = 3, 5, 7, 11$, the found $D_{\max}^{\mathrm{SQP}}$ values agree with the solutions found by BLR-CMA-ES. BLR-CMA-ES robustly finds putative maxima for all $N$. In all instances, it always converges to the same solutions (up to a numerical accuracy of 5 digits). Such robust performance is not observed when using IPOP-CMA-ES or LR-CMA-ES. In fact, these two restart strategies fail to find the putative optima for $N > 11$. This observation suggests that the landscape has a multi-funnel topology where the putative global

maximum is located in a small funnel. The optima found by BLR-CMA-ES suggest a linear relationship between $N$ and $D_{\max}(N)$ with a best linear fit of $D_{\max}^{\text{CMA}} = 0.3251N - 0.04013$ (dotted line in Fig. 7.8).

**Extremal shapes.** The corresponding extremal shapes are depicted in Fig. 7.9. For odd $N$,

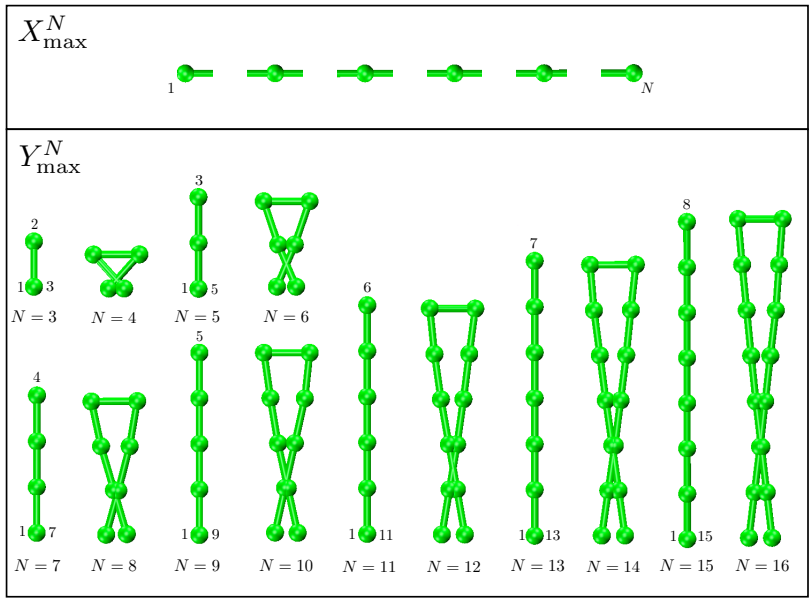



Figure 7.9: The extremal pairs $(X_{\max}^N, Y_{\max}^N)$ for $N = 3, \ldots, 16$ found by the BLR-CMA-ES. The upper box shows the extended shapes $X_{\max}^N$. They assume a linear rod of length $(N-1)b$. The lower box depicts the correspondng shapes $Y_{\max}^N$. For odd $N$, $Y_{\max}^N$ assumes a linear rod shape of half the length of the extended one, where beads $\frac{N+3}{2}$ to $N$ are consecutively folded back onto beads $\frac{N-1}{2}$ to 1. For even $N$, $Y_{\max}^N$ is a planar hairpin where the links from beads $\frac{N+2}{2}$ to $N$ cross the links from beads $\frac{N}{2}$ to 1.

the extremal conformations follow a specific geometric pattern: one structure always is the fully extended linear rod. The other structure is a linear rod with half the length of the extended one, where beads $\frac{N+3}{2}$ to $N$ are folded back onto beads $\frac{N-1}{2}$ to 1. For even $N$, one extremal structure is again the fully extended linear rod. The other extremal structure is a *planar* hairpin with crossed endings. We summarize further numerical data in Table 7.2. For odd $N$, the ACC of the extremal shapes is virtually 0, for even $N$ it is $< 10^{-3}$.

**Landscape visualization.** For $N = 3$ we visualize the full and for $N = 4$ the partial RW-MAX-RMSD landscapes in Fig. 7.10. The 3-bead case is a 2D problem that exhibits a smooth double-funnel landscape with two identical maxima (Fig. 7.10a): either $X$ is the linear rod and $Y$ the folded hairpin, or vice versa. Both BLR-CMA-ES and SQP can solve this problem. The 4-bead case is a 6D problem that cannot be fully visualized. Therefore, we fix one of

| N | $D(N)^{\mathrm{PRS}}_{\mathrm{max}}$ | $D(N)^{\mathrm{CMA}}_{\mathrm{max}}$ | $D(N)_{\mathrm{max}}$ | $\hat{D}(N)_{\mathrm{max}}$ | $ACC(X^N_{\mathrm{max}}, Y^N_{\mathrm{max}})$ |
|---|---|---|---|---|---|
| 3 | 0.9254 | 0.9428 | 0.9428 | 0.9682 | $< 10^{-15}$ |
| 4 | 1.1843 | 1.2360 | - | 1.2909 | $1.15 \cdot 10^{-13}$ |
| 5 | 1.4335 | 1.6 | 1.6 | 1.6137 | $< 10^{-15}$ |
| 6 | 1.6440 | 1.8488 | - | 1.9364 | $8.92 \cdot 10^{-11}$ |
| 7 | 1.9022 | 2.2497 | 2.2497 | 2.2592 | $< 10^{-15}$ |
| 8 | 2.1024 | 2.4978 | - | 2.5819 | $4.43 \cdot 10^{-6}$ |
| 9 | 2.4251 | 2.8974 | 2.8974 | 2.9047 | $< 10^{-15}$ |
| 10 | 2.4609 | 3.0602 | - | 3.227 | $5.68 \cdot 10^{-8}$ |
| 11 | 2.6817 | 3.5443 | 3.5443 | 3.5502 | $< 10^{-15}$ |
| 12 | 2.8877 | 3.8540 | - | 3.8729 | $1.18 \cdot 10^{-7}$ |
| 13 | 2.9992 | 4.1907 | 4.1907 | 4.1957 | $< 10^{-15}$ |
| 14 | 3.2330 | 4.4151 | - | 4.5184 | $1.36 \cdot 10^{-3}$ |
| 15 | 3.3521 | 4.8369 | 4.8369 | 4.8412 | $< 10^{-15}$ |

Table 7.2: The maxima $D_{\mathrm{max}}(N)$ found by PRS and BLR-CMA-ES are summarized in the first two columns. The values of the conjectured analytical $D_{\mathrm{max}}(N)$ for odd $N$ is given in the third column, followed by its asymptotic upper bound (Eq. (7.38)). In the last column we report the ACC values of the extremal shapes.

the extremal shapes to the linear rod $X^4_{\mathrm{max}}$ and compute RMSD's with respect to this shape. We plot the 3 internal angles of $Y$ and color-code the resulting $D(X^4_{\mathrm{max}}, Y)$. This landscape also appears smooth with a single global maximum at $\cos(\theta^Y_1) = \cos(\theta^Y_2) = 2/3, \quad \omega^{\mathrm{Y}}_1 = 0$. All BLR-CMA-ES runs converge to this putative global maximum, while SQP fails in all cases. We conduct additional SQP experiments on the presented simplified landscape with fixed $X^4_{\mathrm{max}}$. We start 1000 SQP runs at random starting positions with the following *fmincon* options: *options = optimset('MaxIter',1000,'RelLineSrchBnd',0.01,'TolCon',1e-12).* The best solutions found during this search are indicated in Fig. 7.10b by circles ($\circ$). In more than 90% of the cases, SQP converges to the local maximum at $\cos(\theta^Y_1) = \cos(\theta^Y_2) = 1$ and arbitrary $\omega^{\mathrm{Y}}_1$. This represents a shape, where all links are completely folded back onto themselves, i.e., the most compact shape. The RMSD between this shape and the linear rod is 1, which is smaller than the putative optimal $D(N)_{\mathrm{max}} = 1.2360$.

### 7.3.4 The RW-MAX-RMSD conjecture

Our numerical data suggest that the extremal shapes for odd $N$ follow a simple geometric pattern: One structure is the fully extended linear rod, the other one a linear fold-back of half length. We thus conjecture that these shapes are extremal *for all odd N*. Under this assumption we propose and derive a general formula for $D_{\mathrm{max}}(N)$ for odd $N$. Combining Eqs. (7.20) and 7.21), we find:

$$\begin{aligned} D^2_{\mathrm{max}}(N) &= D^2(X^N_{\mathrm{max}}, Y^N_{\mathrm{max}}) \\ &= R^2_G(X^N_{\mathrm{max}}) + R^2_G(Y^N_{\mathrm{max}}) - 2ACC(X^N_{\mathrm{max}}, Y^N_{\mathrm{max}}) R_G(X^N_{\mathrm{max}}) R_G(Y^N_{\mathrm{max}}) \\ &= R^2_G(X^N_{\mathrm{max}}) + R^2_G(Y^N_{\mathrm{max}})\,. \end{aligned} \tag{7.27}$$

The above statement for $D^2_{\mathrm{max}}(N)$ is true if we can prove that $ACC(X^N_{\mathrm{max}}, Y^N_{\mathrm{max}}) = 0$ for all odd $N$. Together with analytic formulae for the radii of gyration of $X^N_{\mathrm{max}}$ and $Y^N_{\mathrm{max}}$ we then arrive at a closed-form solution for $D^2_{\mathrm{max}}(N)$.

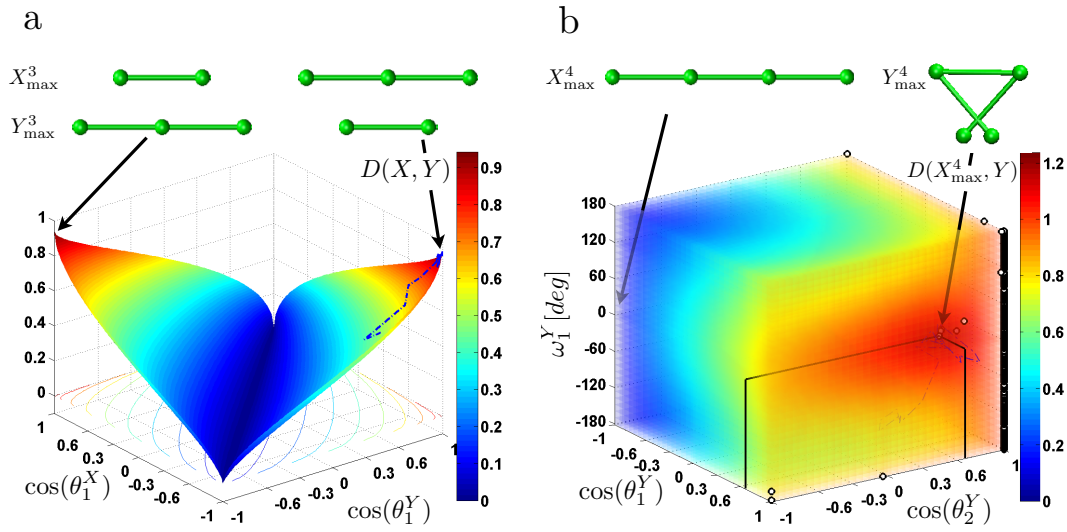



Figure 7.10: a. RW-MAX-RMSD landscape for $N = 3$ with internal angles vs. $D(X, Y)$. The structures above the landscape are the two pairs of extremal shapes. The blue dashed line shows an example evolution path of BLR-CMA-ES. The central ridge with $D(X, Y) = 0$ corresponds to all possible identical structures. b. RW-MAX-RMSD landscape for $N = 4$ with internal representation of $Y$ vs. $D(X^4_{\max}, Y)$, i.e., the first extremal shape is fixed to the linear rod and all RMSD's are calculated with respect to this shape. The hairpin structure corresponds to $\cos(\theta_1^Y) = \cos(\theta_2^Y) = 2/3$ and $\omega_1^Y = 0$ (black solid lines). The blue dashed trajectory is a BLR-CMA-ES evolution path on the full problem. The white circles are solutions found by SQP on the reduced problem (see main text for details).

**The ACC($\mathbf{X^N_{max}}$, $\mathbf{Y^N_{max}}$) for odd N is 0.** Without loss of generality (w.l.o.g.), we assume that $X^N_{\max}$ is the fully extended shape and $Y^N_{\max}$ the folded one, and that their centers of mass are at $(0, 0, 0)$. For odd $N$, the problem of optimal superposition then reduces to a rotation in the $xy$-plane. We define the $x$-axis to be aligned with $X^N_{\max}$ after optimal superposition. $Y^N_{\max}$ forms a certain rotation angle $\alpha$ with $X^N_{\max}$ as shown in Fig. 7.11. In order to see that for the specific pair of configurations $(X^N_{\max}, Y^N_{\max})$ the $ACC(X^N_{\max}, Y^N_{\max})$ is 0 for any rotation angle $\alpha$ and any odd $N$, we recall the definition of the aligned correlation coefficient for two optimally aligned structures $X$, $Y$:

$$ACC(X,Y) = \frac{\sum_{i=1}^{N} \mathbf{x}^{(i)} \cdot \mathbf{y}^{(i)}}{\sqrt{\sum_{i=1}^{N} \mathbf{x}^{(i)2} \sum_{i=1}^{N} \mathbf{y}^{(i)2}}}\,. \tag{7.28}$$

The denominator of this expression must always be positive because the two factors in the square root are the sum of the squared atomic coordinates of the two extremal shapes.

From Fig. 7.11 we see that the coordinate vectors $x^{(i)}_{\max}$ only have non-zero entries in $x$ direction. Furthermore, the $x$ coordinate of the $i^{\text{th}}$ bead in $X^N_{\max}$ is the negative of the $x$ coordinate of the $(N - i + 1)^{\text{th}}$ bead. The central bead (i.e. the $\frac{N+1}{2}{}^{\text{th}}$ bead) in $X^N_{\max}$ is at $(0, 0, 0)$, so the

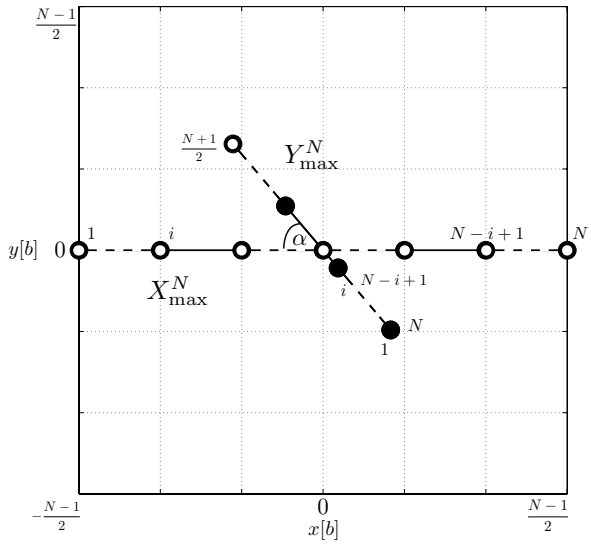



Figure 7.11: General setup for the calculation of the RMSD between $X^N_{\max}$ and $Y^N_{\max}$ for odd $N$ after optimal superposition. $X^N_{\max}$ is the extended structure and $Y^N_{\max}$ is the folded structure. Open circles ($\circ$) represent positions that are occupied by single beads, filled circles ($\bullet$) indicate positions occupied by two beads. The two structures enclose a planar angle $\alpha$.

scalar product with its corresponding bead in $Y^N_{\max}$ will be 0. The positions of the $i^{\text{th}}$ and the $(N-i+1)^{\text{th}}$ bead in $Y^N_{\max}$ are identical (filled circles in Fig. 7.11), independent of the angle $\alpha$. The numerator in $ACC(X^N_{\max}, Y^N_{\max})$ thus becomes:

$$\begin{aligned}\sum_{i=1}^{N} \mathbf{x}^{N,(i)}_{\max} \cdot \mathbf{y}^{N,(i)}_{\max} &= \sum_{i=1}^{\frac{N+1}{2}-1} \mathbf{x}^{N,(i)}_{\max} \cdot \mathbf{y}^{N,(i)}_{\max} + 0 + \sum_{i=\frac{N+1}{2}-1}^{N} \mathbf{x}^{N,(i)}_{\max} \cdot \mathbf{y}^{N,(i)}_{\max} \\ &= -\sum_{i=\frac{N+1}{2}-1}^{N} \mathbf{x}^{N,(i)}_{\max} \cdot \mathbf{y}^{N,(i)}_{\max} + \sum_{i=\frac{N+1}{2}-1}^{N} \mathbf{x}^{N,(i)}_{\max} \cdot \mathbf{y}^{N,(i)}_{\max} \\ &= 0\end{aligned} \tag{7.29}$$

and, therefore, $ACC(X^N_{\max}, Y^N_{\max}) = 0$ for all odd $N$ and any rotation angle $\alpha$.

**The radii of gyration of $\mathbf{X^N_{max}}$ and $\mathbf{Y^N_{max}}$ for odd N.** We assume w.l.o.g. that $X^N_{\max}$ is aligned along the $x$-axis and $Y^N_{\max}$ along the $y$-axis. Computing $D^2_{\max}(N)$ then reduces to the calculation of two one-dimensional, $N$-dependent radii of gyration (Eq. (7.27)). For $X_{\max}$ we

find:

$$\begin{aligned} R_G^2(X_{\max}^N) &= \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_{\max}^{N,(i)} - \mathbf{x}_{\mathrm{cm,max}}^N\|^2 = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_{\max}^{N,(i)}\|^2 \\ &= \frac{1}{N}\sum_{i=0}^{N-1}\left(-\frac{N-1}{2}b + ib\right)^2 = \frac{2}{N}b^2\sum_{i=1}^{\frac{N-1}{2}}(i)^2 . \end{aligned} \tag{7.30}$$

Defining $M^- = \frac{N-1}{2}$ yields the result:

$$R_G^2(X_{\max}^N) = \frac{2}{N}b^2\sum_{i=1}^{M^-}(i)^2 . \tag{7.31}$$

For $Y_{\max}$ we assume w.l.o.g. that the first bead is located at the origin (translation invariance of $R_G$). We then find:

$$\begin{aligned} R_G^2(Y_{\max}^N) &= \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{y}_{\max}^{N,(i)} - \mathbf{y}_{\mathrm{cm,max}}^N\|^2 \\ &= -\mathbf{y}_{\mathrm{cm,max}}^N \cdot \mathbf{y}_{\mathrm{cm,max}}^N + \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{y}_{\max}^{N,(i)}\|^2 . \end{aligned} \tag{7.32}$$

The center of mass of $Y_{\max}^N$ is at $y_{\mathrm{cm,max}}^N = \frac{N-1}{N}\frac{N-1}{4}b$ (McLachlan, 1984), thus:

$$\begin{aligned} R_G^2(Y_{\max}^N) &= -\left(\frac{(N-1)(N-1)}{4N}b\right)^2 + \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{y}_{\max}^{N,(i)}\|^2 \\ &= -\left(\frac{(N-1)(N-1)}{4N}b\right)^2 + \frac{1}{N}\left(\left(\frac{N-1}{2}b\right)^2 + 2\sum_{i=0}^{\frac{N-3}{2}}(ib)^2\right) . \end{aligned} \tag{7.33}$$

Defining $\hat{M}^- = \frac{N-3}{2}$, we find the result:

$$R_G^2(Y_{\max}^N) = -b^2\left(\frac{M^-M^-}{N}\right)^2 + \frac{1}{N}b^2\left((M^-)^2 + 2\sum_{i=1}^{\hat{M}^-}(i)^2\right) . \tag{7.34}$$

**Analytical form of $D_{\max}^2(N)$.** Combining Eqs. (7.31) and 7.34 we arrive at the final formula:

$$D_{\max}^2(N) = \frac{2}{N}b^2\sum_{i=1}^{M^-}(i)^2 - b^2\left(\frac{M^-M^-}{N}\right)^2 + \frac{1}{N}b^2\left((M^-)^2 + 2\sum_{i=1}^{\hat{M}^-}(i)^2\right) . \tag{7.35}$$

The values of $D_{\max}^2(N)$ for $N = 3, \ldots, 15$ are tabulated in Table 7.2. Note that the complicated form of $R_G^2(Y_{\max}^N)$ comes from the fact that the mass distribution in $Y_{\max}^N$ is slightly asymmetric

(the position of the center bead at the turn is occupied only once) and hence the center of gravity is not located in the middle between the first and the $\frac{N+1}{2}^{\text{th}}$ bead. This asymmetry, however, decreases with increasing $N$, and $D_{\max}(N)$ is asymptotically bounded by a simpler expression.

**Asymptotic bound of $\mathbf{D_{max}(N)}$ for odd N.** In order to study the asymptotic behavior of $D_{\max}(N)$ for large odd $N$, we define the ratio

$$C_{\max}(N)^2 = \frac{D^2_{\max}(N)}{N^2} = \frac{R^2_G(X^N_{\max})}{N^2} + \frac{R^2_G(Y^N_{\max})}{N^2}. \tag{7.36}$$

Using Eq. (7.31), we find for the first summand:

$$\begin{aligned}
\frac{R^2_G(X^N_{\max})}{N^2} &= \frac{2}{N^3} b^2 \sum_{i=1}^{M^-} (i)^2 = \frac{2}{N^3} b^2 \left( \frac{M^-}{6} (2M^- + 1)(M^- + 1) \right) \\
&= \frac{2}{N^3} b^2 \left( \frac{N-3}{12} \left( 2 \left( \frac{N-3}{2} \right) + 1 \right) \left( \left( \frac{N-3}{2} \right) + 1 \right) \right) \\
&= \frac{2}{N^3} b^2 \left( \frac{N^3}{24} + \mathcal{O}\left(N^2\right) \right) = \frac{1}{12} b^2 + \mathcal{O}\left( \frac{1}{N} \right) b^2 ,
\end{aligned}$$

and using Eq. (7.34) for the second summand:

$$\begin{aligned}
\frac{R^2_G(Y^N_{\max})}{N^2} &= -b^2 \left( \frac{M^- M^-}{N^2} \right)^2 + \frac{1}{N^3} b^2 \left( (M^-)^2 + 2 \sum_{i=0}^{\hat{M}^-} (i)^2 \right) \\
&= -\frac{1}{16} b^2 + \mathcal{O}\left( \frac{1}{N^2} \right) b^2 \\
&+ \frac{1}{N^3} b^2 \left( (M^-)^2 + 2 \left( \frac{\hat{M}^-}{6} (2\hat{M}^- + 1)(\hat{M}^- + 1) \right) \right) \\
&= -\frac{1}{16} b^2 + \mathcal{O}\left( \frac{1}{N^2} \right) b^2 + \frac{1}{12} b^2 + \mathcal{O}\left( \frac{1}{N} \right) b^2 \\
&= -\frac{1}{16} b^2 + \frac{1}{12} b^2 + \mathcal{O}\left( \frac{1}{N} \right) b^2 = \frac{1}{48} b^2 + \mathcal{O}\left( \frac{1}{N} \right) b^2 .
\end{aligned}$$

$C_{\max}(N)^2$ thus has the asymptotic limit:

$$\lim_{N \to \infty} C_{\max}(N)^2 = \frac{1}{12} b^2 + \frac{1}{48} b^2 = \frac{5}{48} b^2 . \tag{7.37}$$

We can thus derive an asymptotic upper bound $\hat{D}_{\max}(N)$ for large $N$:

$$D_{\max}(N) \lesssim \hat{D}_{\max}(N) = \sqrt{\frac{1}{12} + \frac{1}{48}} bN = \sqrt{\frac{5}{48}} bN = \frac{1}{4} \sqrt{\frac{5}{3}} bN . \tag{7.38}$$

This result relates to the moment of inertia $I_{\mathrm{cm}}(X)$ of a linear rod $X$ of length $L$ and mass $m$ around its center of mass:

$$I_{\mathrm{cm}}(X) = \frac{1}{12} m L^2 \,. \tag{7.39}$$

Using $m = 1$ for both $X^N_{\max}$ and $Y^N_{\max}$, a length of $(N-1)b$ for $X^N_{\max}$, and a length of $\frac{1}{2}(N-1)b$ for $Y^N_{\max}$ confirms the result in Eq. (7.37) asymptotically for large $N$:

$$I_{\mathrm{cm}}(X^N_{\max}) = \frac{1}{12}\left((N-1)b\right)^2 \approx \frac{1}{12} N^2 b^2 \,, \tag{7.40}$$

$$I_{\mathrm{cm}}(Y^N_{\max}) = \frac{1}{12}\left(\frac{1}{2}(N-1)b\right)^2 \approx \frac{1}{48} N^2 b^2 \,. \tag{7.41}$$

The values $\hat{D}_{\max}(N)$ are also tabulated in Table 7.2. As the relative error $\eta = \frac{|D_{\max}(N) - \hat{D}_{\max}(N)|}{|D_{\max}(N)|}$ already drops below 1% for $N \geq 5$, $\hat{D}_{\max}$ is a good estimate of the upper bound for odd $N$. Furthermore, we conjecture that $\hat{D}_{\max}$ is also an asymptotic upper bound for even $N$. The reasoning is twofold: (i) visual inspection of the extremal shapes for even $N$ suggests that for large $N$ the folded structure will assume an "odd-like" conformation where all except the central links are co-linear. (ii) already for even $N \geq 10$ the relative error between $\hat{D}_{\max}$ and $D^{\mathrm{CMA}}_{\max}$ drops below 1% (Fig. 7.8, dotted line).

### 7.3.5 The maximum RMSD problem for self-avoiding Random Walks

Our previous investigations conjecture an upper bound for the RMSD-based landscape diameter of linear chains with fixed link length $b$ and without any restrictions on the degrees of freedom. It is, however, clear that the hairpin-like extremal shape can never be attained by any real chain molecule since $N - 1$ beads overlap. We therefore study the effect of self-avoidance on the extremal shapes, where beads are not allowed to occupy the same position in space. The resulting *Self-avoiding Walk Maximum RMSD problem* (SAW-MAX-RMSD) can be stated analogous to RW-MAX-RMSD, but with additional quadratic constraints on the allowed distances between any two beads. We denote again the extremal shapes by ($X^N_{\max}$, $Y^N_{\max}$) with $N$ beads that maximize the minimal squared RMSD $D^2(X,Y)$ over all possible pairs of $X$ and $Y$:

$$(X^N_{\max}, Y^N_{\max}) = \arg\max_{\mathrm{X,Y}} D^2(X,Y) = \arg\max_{\mathrm{X,Y}} \min_{\mathbf{R}} \frac{1}{N} \|\mathbf{R}X_0 - Y_0\|^2 \,. \tag{7.42}$$

$X, Y$ represent now SAW's. The degree of self-avoidance is controlled by the parameter $c_{\mathrm{r}}$ that represents the radius of a sphere around each bead that cannot be penetrated by any the sphere of any other bead (see Fig. 7.1 for a sketch). As in the finite sphere packing problem, these constraints render the landscape discontinuous. In order to arrive at a feasible black-box optimization formulation we augment the objective function $f(\hat{\boldsymbol{q}}_{\mathrm{S}})$ in Eq. (7.26) with an energy $E_{\mathrm{modLJ}}(X,Y)$ that penalizes overlapping configuration in a soft manner. $E_{\mathrm{modLJ}}(X,Y)$ is defined as

$$E_{\mathrm{modLJ}}(X,Y) = \sum_{X,Y} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} u_{\mathrm{modLJ}}(r_{ij}) \,, \tag{7.43}$$

where $r_{ij}$ is the distance between the $i^{\text{th}}$ and $j^{\text{th}}$ bead. The modified LJ pair potential is

$$u_{\text{modLJ}}(r_{ij}) = \begin{cases} \epsilon + u_{\text{LJ}}(r_{ij}) & \text{if } r_{ij} < 2c_{\text{r}} \\ 0 & \text{else} \end{cases} \tag{7.44}$$

with $\epsilon = 1$ and $\sigma_{\text{LJ}} = 2^{-1/6}$. This implies that the penalizing energy term only vanishes if all constraints are satisfied. The resulting black-box objective function $f_{\text{SAW}}$ to be maximized for SAW-MAX-RMSD then reads:

$$f_{\text{SAW}}(\hat{\boldsymbol{q}}_{\text{S}}) = f(\hat{\boldsymbol{q}}_{\text{S}}) - E_{\text{modLJ}}(J(T(\hat{\boldsymbol{q}}_S))\,. \tag{7.45}$$

Because the described penalty approach uses a soft potential, it cannot guarantee that the resulting extremal shapes exactly satisfy all self-avoidance constraints. This has to be ensured *a posteriori* by inspecting the geometries of the putative extremal shapes.

### 7.3.6 Numerical solutions of SAW-MAX-RMSD

Due to its robust performance on RW-MAX-RMSD we also apply BLR-CMA-ES to this problem. The identical simulation protocol is used as for RW-MAX-RMSD. We consider 5 different SAW's: $c_{\text{r}} = 0b, 0.025b, 0.125b, 0.375b, 0.495b$. The zero value is included in order to validate the penalty approach; it should yield identical results to RW-MAX-RMSD. The highest value is chosen such that beads connected by a link of length $b$ do not repel each other. We present the identified putative upper bounds $D_{\max}(N)$ and the ratios $C_{\max}(N) = D_{\max}(N)/N$ in Fig. 7.12. BLR-CMA-ES always converges to solutions that agree to within an objective value

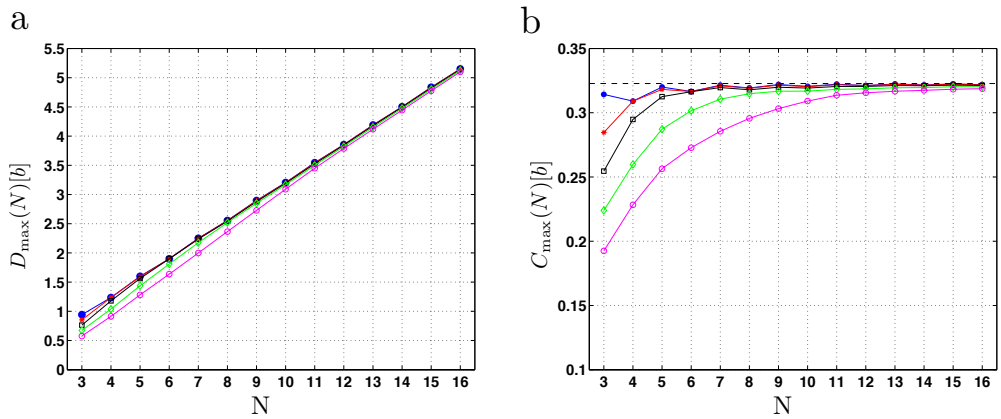



Figure 7.12: a. Solutions of SAW-MAX-RMSD identified by BLR-CMA-ES for $c_{\text{r}} = 0b$ (blue •), $0.025b$ (red $*$), $0.125b$ (black □), $0.375b$ (green ◇) and $0.495b$ (pink ○). b. Ratio $C_{\max}(N) = D_{\max}(N)/N$ for better visual discrimination. The horizontal dashed line is the asymptotic limit $\hat{D}_{\max}(N)/N = \sqrt{5/48}$.

of $10^{-3}$ for any $c_{\text{r}}$ and $N$. For all reported putative optimal solutions the corresponding extremal structures satisfy all self-avoidance constraints. This suggest that our penalty function

$E_{\mathrm{modLJ}}(X, Y)$ works effectively. The results for $c_{\mathrm{r}} = 0b$ are identical to those from the original RW-MAX-RMSD problem. All results are consistent in the sense that increasing $c_{\mathrm{r}}$ for a given $N$ lowers the putative upper bound. The effect of self-avoidance is especially apparent in the cases with $N < 10$. With increasing $N$, the solutions of SAW-MAX-RMSD rapidly approach the conjectured RMSD landscape diameter of RW's. This is also reflected in the asymptotic behavior of the ratio $C_{\max}(N)$ (Fig. 7.12b). The difference in the upper bounds is contributed solely by the hairpin-like extremal configuration, as the linear rod naturally satisfies self-avoidance for all $c_{\mathrm{r}}$. Visual inspection of the hairpins provides a coherent picture of how increasing self-avoidance changes the geometry of the hairpin. An example is shown in Fig. 7.13 for $N = 12$. While the effect of self-avoidance has little influence on the turn of the

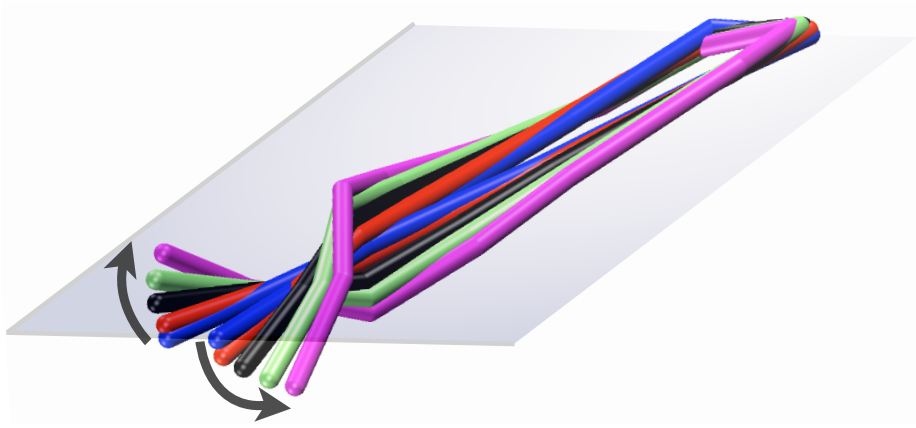


Figure 7.13: Effect of self-avoidance on the extremal hairpin structure for $N = 12$. The crossing at the lower end opens up and forms a dual helix (Banavar et al., 2009). The color-code is identical to that of Fig. 7.12 and indicates the $c_{\mathrm{r}}$ values used.

hairpin, the crossing of the lower ends changes to a winding of the first and last four beads. Banavar and co-workers called a hairpin with complete winding a *dual helix* (Banavar et al., 2009).

### 7.3.7 A comparison of extremal shapes and protein structural motifs

All globular proteins are composed of a number of structural motifs, i.e., small regular geometric patterns whose unique combination gives rise to the structural diversity of the protein space. The most common motifs (or secondary structures) are the $\alpha$-helix, the $3_{10}$-helix, the poly-proline type II (ppII) helix, and the parallel (para) and anti-parallel (anti) $\beta$-strand. Their shapes, along with the $C_\alpha$-pseudo-angle description, are depicted in Fig. 7.14. The simplest super-secondary structure motif is the hairpin. It consists of two helices or strands linked by a turn. We note that the identified extremal shapes resemble some of these naturally occurring geometries. In order to quantify these observation we conduct the following experiment. We first identify the extremal shapes for protein-like SAW's. The difference to the previous extremal shapes is a further restriction of the $\theta_i$-angles. Oldfield and Hubbard

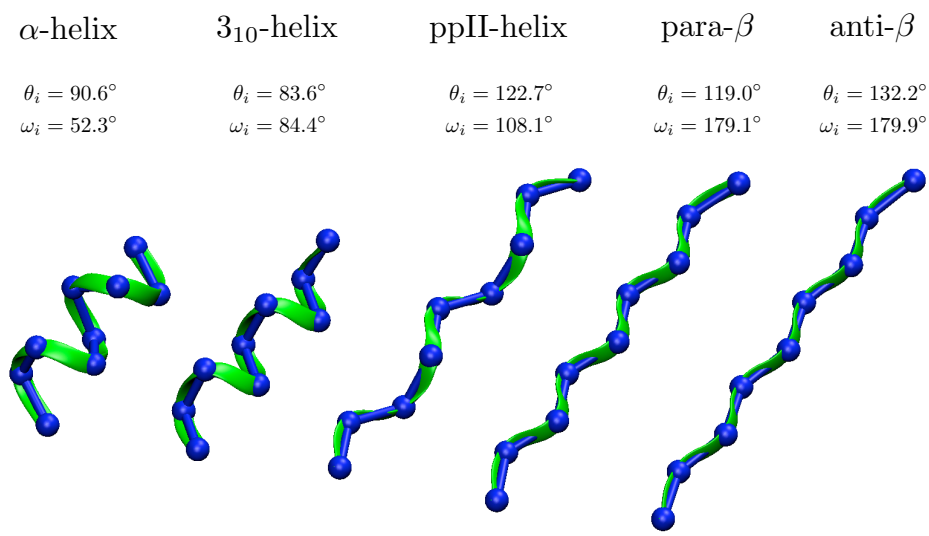



Figure 7.14: $C_\alpha$ backbones of the five most common structural motifs along with their pseudo-angle description.

analyzed the protein space in terms of $C_\alpha$ geometry and presented bounds for the $\theta$ angles from a large set of protein crystal structures (Oldfield and Hubbard, 1994). They concluded that in real proteins $\theta_i \in [85, 145]$. Together with the self-avoidance parameter $c_\mathrm{r} = 0.445$ we derived the extremal shapes $(X^N_\mathrm{max}, Y^N_\mathrm{max})$ under these constraints using our optimization protocol. The link length is set to $b = 3.8$ and interpreted in units of Ångström. This is the average distance between $C_\alpha$ atoms in proteins.

The resulting extremal shapes are similar to the previously described SAW's. The linear rod, however, is changed to a zig-zag structure with $\theta_i = 145^o$. We focus here on the case $N = 12$. We generate an ensemble of 3000 protein-like 12-bead SAW's as a base sample of the conformational space. We include the structural motifs and the extremal shapes. As an example of a hairpin motif of length 12 we consider a Tryptophan zipper (trypzip) ensemble (PDB code: 1LE1) (Cochran et al., 2001), which comprises 20 $\beta$-hairpin structures. We calculated all-against-all RMSD's for this ensemble and store the data in a distance matrix. A convenient way to visualize this conformational ensemble is offered by low-dimensional embedding techniques. We use MATLAB's Multi-Dimensional-Scaling (MDS) function (*mdscale.m*) with standard setting. MDS attempts to find an arrangement of points in a lower-dimensional space such that all calculated pairwise distances are (approximately) conserved. We present a 2D MDS embedding of the ensemble in Fig. 7.15. We note that the 2D projection of the conformational landscape is more extended in one coordinate than in the other. The scalar $x_1^\mathrm{MDS}$ coordinate is spanned by our upper bound ($< \sqrt{5/48} \cdot 3.8 \cdot 12$ Å$= 14.7$Å). In $x_2^\mathrm{MDS}$, maximum distances are around 7 Å. The overall shape of the ensemble is pear-like with more structures neighboring the extended extremal shape. We observe that (i) the $\beta$ strands are extremely similar to the extended extremal shape (RMSD $< 1$Å) and (ii) for all structures in

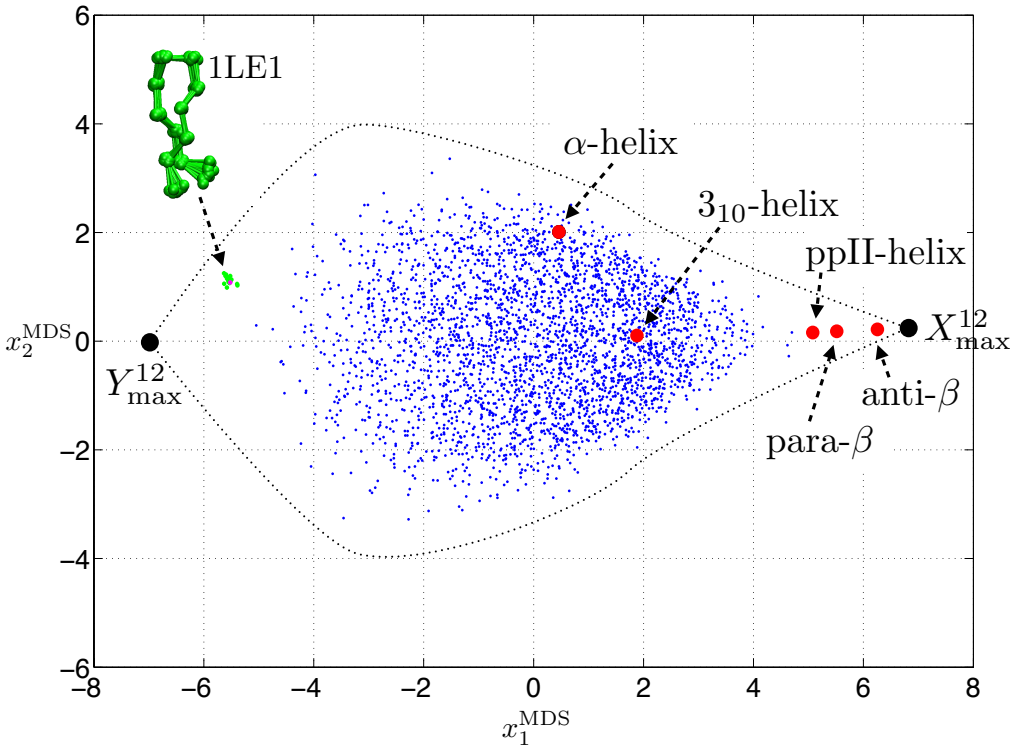



Figure 7.15: 2D MDS embedding of a protein-like SAW ensemble (see main text for details). $\beta$ strands are close to the extended extremal shape $X^{12}_{\max}$ (RMSD $< 1$ Å). The 1LE1 hairpin ensemble is neighboring the hairpin-like extremal shape (RMSD $\approx 2$ Å). The dotted line sketches the overall shape of the point cloud.

the ensemble the trypzip ensemble is closest to the hairpin-like extremal shape with RMSD $\approx 2$Å. These results support the hypothesis that real protein structural motifs span a large portion of the entire conformation space when the RMSD metric is used as distance measure.

## 7.4 Conclusions

The local shape probability distribution within the RW ensemble (i.e., the local density of structures in the immediate neighborhood of a given central structure, relative to its average over all possible central structures) is by no means homogeneous across all possible shapes. Even in the absence of interatomic interactions (beyond the mere connectivity constraint), some shapes are intrinsically more probable, while others (e.g., those defined by a central structure with one or more bond angles equal to 0 or $\pi$) have a vanishing probability. Over the limited range of sizes ($N = 3, \ldots, 6$) that could be probed in the present study, the bias

in favor of the most probable (densest) shape increases in the sequence 1.28, 1.79, 2.94, and 10.05, as measured by the probability of this shape relative to the corresponding average over all possible shapes. In other words, given a random structure with $N = 6$ beads and prompted to make a guess for a shape to which this structure belongs, one would have about a tenfold higher chance of success by proposing the densest shape than by proposing any shape at random. Over the range $N = 3, \ldots, 6$, the central structures associated with the densest shape were all found to be planar, with bond angles progressively opening with increasing $N$ (Fig. 7.7). Based on the present results, it is, however, not possible to determine whether these features also hold for larger $N$.

The finite-cutoff shape probability distribution (i.e., the integrated density of structures within a specified cutoff distance from a given central structure, relative to its average over all possible central structures) as well as the fractional coverage function (i.e., the integrated density of structures within a specified finite cutoff distance from a given central structure, relative to its value at infinite cutoff) evidence similar qualitative features for all values of $N$ considered.

Three regimes are observed upon increasing the cutoff $R_c$ : (i) for $R_c$ below a threshold value $R_N^*$, all shapes only encompass part of the ensemble; (ii) for $R_c$ above a threshold value $R_N^{**} > R_N^*$, all shapes encompass the entire ensemble; (iii) for intermediate values $R_N^* \leq R_c \leq R_N^{**}$, a single shape ($R_c = R_N^*$; barycentric shape), and then an increasingly larger set of shapes ($R_c > R_N^*$), encompasses the entire ensemble.

Over the range $N = 3, \ldots, 6$, the central structures associated with the barycentric shapes are not planar (except for $N = 3$) and do not appear to present obvious systematic trends (Fig. 7.7). This may, however, also be due to the relatively low precision in the determination of these structures for $N = 5$ and 6.

The observed threshold $R_N^{**}$ has led us to investigate the $N$-dependent landscape diameter of linear chains under RMSD. We have combined numerical optimization experiments and analytical geometry in a first attempt to derive a tight RMSD upper bound. The regular geometric pattern of the extremal shapes for odd $N$ up to 15 inspired a conjecture about an analytic formula for the upper bound of the RMSD of RW's. An conjectured asymptotic formula for large $N$ also holds for SAW's.

The numerically obtained pairs of extremal conformations for RW's and SAW's reveal that one extremal structure is a linear rod and the other one is hairpin-like. When adopting angle restrictions from real protein structures, the extremal shapes show a remarkable similarity to naturally occurring structural motifs, such as $\beta$-strands and $\beta$-hairpins. This suggests that proteins span a large portion of the possible conformation space when RMSD is used to measure distances.

# 8

# Conclusion & Future Work

> "Kids, you tried your best and you failed miserably. The lesson is: Never try."
> Homer Simpson, in: The Simpsons, Burn's Heir, Episode no. 99, 1994

This thesis was dedicated to the characterization, optimization, and sampling of black-box landscapes. We then applied these tools to the geometric configurations of atomic clusters and linear chain molecules in different contexts. In this chapter, we summarize the key conclusions that can be drawn from the presented studies and outline a number of possible future research opportunities.

**Landscapes.** We started this thesis by revisiting the landscape concept in physics, chemistry, biology, and optimization. To the best of our knowledge, there has been no previous attempt to review this paradigm across all these disciplines. The landscape paradigm has been vital for the analysis of black-box systems presented in this thesis. We conclude that popularizing the available knowledge across disciplines might foster interdisciplinary collaborations between theoretical physicists, theoretical biologists, and computer scientists.

**Characterization.** We have introduced a set of statistical measures that allow characterizing continuous black-box landscapes with respect to: global landscape topology, variable importance and separability, and landscape ruggedness. Fitness-distance correlations and dispersion differences have been used to study the global landscape topology. Variable importance and separability have been probed with a modified Morris method and two derived, dimension- and scale-independent scalar quantities: the normalized total importance variation $t_\mu$ and the normalized total interaction variation $t_\sigma$. The measure of ruggedness is based on the random

walk autocorrelation of objective function values and the derived correlation length.

All landscape descriptors have been tested on the CEC benchmark test suite with standard FES budget. We found that fitness-distance correlation and dispersion differences can discriminate between functions with single-funnel topology and highly unstructured problems. However, high-conditioned ellipsoidal functions, although convex and unimodal, cannot be distinguished from multi-funnel problems by any of the measures. This observation is in stark contrast to Lunacek and Whitley's claim that function dispersion would be a good measure for detecting uni-modality (Lunacek and Whitley, 2006). The observed problems with high-conditioned (globally) ellipsoidal landscapes may be alleviated by considering alternative distance metrics. Replacing the Euclidian distance by a suitable Mahalanobis distance that captures the underlying ellipsoid might increase the discriminative power of fitness-distance correlation and dispersion. The required positive definite matrix for the Mahalanobis distance might be inferred from the covariances learned by CMA-ES and Gaussian Adaptation.
The Morris method and its derived quantities can identify separable functions across all tested dimensions, but also produce false-positive results, for instance on the multi-modal problem $f_{14}$. The estimated landscape correlation length from the random walk autocorrelation was found to discriminate between smooth quadratic functions and most multi-modal or noisy functions. Several highly multi-modal landscapes, however, also exhibit a large correlation length thus limiting the discriminative power of this measure as well.

From the observed limitations we conclude that the present study can only be considered a first step toward more principled statistical landscape characterization. We envision an unsupervised statistical learning framework in which a classification of black-box landscapes can be inferred from a given list of samples and associated landscape descriptors. We believe that the statistical fingerprints introduced here provide informative features to this end. Equipped with such statistical "problem classes", the ultimate goal would be to relate these classes to a prediction of success performance of state-of-the-art search heuristics.

Our current research efforts also include the application of these landscape descriptors to real-world black-box problems, such as parameter estimation landscapes from systems biology. The possibility of re-using samples from black-box optimization runs for landscape analysis has been and still is a topic of our research (Müller et al., 2007).

**Optimization.** In this thesis we have solely considered variable-metric approaches to black-box optimization in form of the CMA-ES and Gaussian Adaptation. Our study of standard CMA-ES suggest that (i) the initial CEC 2005 benchmark results of IPOP-CMA-ES were partly incorrect due to wrong boundary settings. From the reduced performance of CMA-ES when using correct boundary settings, we conclude that the current boundary handling technique (Hansen et al., 2009) could be improved. (ii) LD sampling is beneficial for the performance of CMA-ES in all tested benchmarks, and (iii) BLR-CMA-ES as an alternative restart strategy is useful for solving problems of the RW-MAX-RMSD type.

In order to improve the performance of CMA-ES on multi-funnel landscapes we introduced the concept of parallel CMA-ES schemes. One such instance developed in this thesis, the

collaborative Particle Swarm CMA-ES, exhibits the best performance among all CMA-ES variants on the CEC 2005 benchmarks in $n = 10$. For all strongly multi-modal and multi-funnel problems, PS-CMA-ES ranks on average better than the standard restart strategies of CMA-ES across all tested dimensions. Parallel CMA-ES schemes thus supplement sequential CMA-ES in the absence of a globally unimodal structure. The simplest combination of both schemes is a restart strategy where, instead of increasing the population size, the number of parallel CMA-ES instances is increased. Such a scheme would combine the strength of single CMA-ES on smooth unimodal landscapes with the robustness of parallel CMA-ES on multi-funnel problems.

Gaussian Adaptation, a largely ignored optimization and design-centering method developed in the late 1960's has been revisited and enhanced. Gaussian Adaptation offers an alternative view on step size adaptation: the maximum entropy principle. For gradually decreasing objective function thresholds, a multivariate normal distribution is adapted for maximum entropy. We introduced Restart Gaussian Adaptation, which shows excellent performance on the CEC 2005 benchmark, comparable to IPOP-CMA-ES. Restart GaA is less favorable than IPOP-CMA-ES on strongly multi-modal functions with globally convex structure, but outperforms all tested methods on noisy unimodal functions. Future work will explore the possibility of including parallelism into the Gaussian Adaptation framework.

From the encouraging performance results on the CEC 2005 benchmark, we conclude that Gaussian Adaptation, CMA-ES (with LD sampling), and PS-CMA-ES are applicable to a wide range of landscape types. Future research will include the application of these methods to real-world problems, such as parameter estimation in systems biology and geometric problems in biophysics and architecture.

**Sampling.** Gaussian Adaptation has also served as a conceptual link to black-box sampling. In fact, by changing the acceptance criterion from a hard threshold to a Metropolis criterion, we derived a novel adaptive Markov-Chain Monte Carlo sampler. This Metropolis Gaussian Adaptation algorithm performs comparably well to the seminal AP algorithm on selected target distributions. We also demonstrated the current limitations of M-GaA on Neal's funnel distribution. Future work will consider the vanishing adaptation concept in order to prove ergodicity of Metropolis Gaussian Adaptation. We will also explore the possibility of (convex) volume estimation using Gaussian Adaptation (see Appendix Section A1.3 for an example). Furthermore, we wish for a closer collaboration between the computational statistics and the black-box optimization communities. Too many ideas have been re-invented in both communities.

**Atomic Clusters.** We proposed the identification of minimum-energy configurations of atomic clusters under different pair potentials as novel benchmarks for black-box optimization. Cluster problems introduce isospectral symmetry as a novel problem characteristic. We introduced Cohn-Kumar clusters, which have provable ground states for certain instance sizes. We showed that these landscapes are much smoother than Lennard-Jones clusters with comparably many degrees of freedom. We also presented a high-dimensional benchmark, the 38-atom Lennard-Jones cluster, with a tunable landscape topology. We conclude that our set of cluster

benchmarks represents a rich test bed that should be included in future black-box benchmark scenarios. This might be achieved by embedding the presented cluster problem instances into the BBOB framework.

**Linear chains.** We analyzed the random walk linear chain model of polymers pertaining to the most important distance measure in structural biology: the Root Mean Square Deviation (RMSD) after least-squares roto-translational fitting. We investigated the resulting conformational landscape in the absence of energetic terms. Two properties of this landscape have been of specific interest: (i) the inhomogeneity of the local neighborhood density induced by the RMSD and (ii) the identification of pairs of structures that have maximum RMSD and thus represent the extremal boundaries of the landscape.

We quantified the inhomogeneity of the local (infinitesimal) neighborhood density (or local shape probability distribution) within ensembles of random walks consisting of up to $N = 6$ beads. We could show that some shapes are intrinsically more probable, while others have a vanishing probability. The bias in favor of the most probable (densest) shape increases in the sequence 1.28, 1.79, 2.94 and 10.05, as measured by the probability of this shape relative to the corresponding average over all possible shapes. These densest shapes were all found to be planar, with bond angles progressively opening with increasing $N$.

The finite-cutoff shape probability distribution evidence similar qualitative features for all value $N$ considered. Three regimes could be observed with increasing cutoff $R_c$. Below a threshold $R_N^*$, the shape probabilities are highly inhomogeneous. For $R_c = R_N^*$, single (or a few symmetry-related) barycentric shapes could be identified that encompass the entire random walk ensemble. These shapes are not planar and do not show a clear geometric pattern. Above the threshold $R_N^{**}$, all shapes encompass the entire random walk ensemble. We argue that the derived shape probabilities constitute useful prior information that should be included in Bayesian approaches toward structure identification.

The structures associated with the threshold $R_N^{**}$ have been further investigated for walks with $N = 3, \ldots, 15$ beads using a combination of BLR-CMA-ES optimization runs and analytic geometry. The putative pairs of extremal conformations reveal that one extremal structure always is a linear rod and the other one is hairpin-like. This regular geometric pattern allowed a conjecture about an analytic formula for the upper bound of the RMSD of random walks. The conjectured asymptotic formula for larger $N$ also holds for self-avoiding walks. Future research will explore possibilities how to prove this conjecture.

When adopting angle restrictions from real protein structures, the extremal shapes show a remarkable similarity with naturally occurring structural motifs, such as the $\beta$-strands and $\beta$-hairpins. This suggests that real proteins span a large portion of the possible conformation space when RMSD is used to measure distances. Future research will be concerned with relating the derived extremal structures with known folding times of naturally occurring protein motifs, for instance, by analyzing their internal contact order.

# Appendix

## A1 GaALib: A MATLAB toolbox for Gaussian Adaptation

The Gaussian Adaptation Library (GaALib) is a set a MATLAB toolbox. We briefly outline the contents of the different functions and scripts.

### A1.1 Algorithm

The core of the library is the MATLAB routine *gaussAdapt.m*. The header of the function is listed below.

```
function [xmin,fmin,counteval,out] = gaussAdapt(fitfun,xstart,inopts)
%-------------------------------------------------------------------------
% Implementation of the Gaussian Adaptation algorithm for design centering
% Black-box optimization and adaptive MCMC sampling
%
% Input:
% fitfun: Name of the fitness/target function as string or function handle
% xstart: initial candidate solution/sample
% inopts: option structure that determines internal strategy parameters
%         (see code for details)
%
% Output:
% xmin: minimum candidate solution found by GaA (when using GaA as optimizer)
% fmin: fitness value of the xmin (when using GaA as optimizer)
% counteval: Number of function evaluations
% out: Output structure storing all relevant information (see code for
% details)
%
%
% Christian L. Mueller
% MOSAIC group, ETH Zurich, Switzerland
%-------------------------------------------------------------------------

% dimension of the problem
N = length(xstart);

% Options defaults: Stopping criteria % (value of stop flag)
defopts.StopFitness   = -Inf;    % stop if f(xmin) < stopfitness, minimization';
defopts.MaxIter       = 1e4*(N); % maximal number function evaluations';
defopts.TolX          = 1e-12;   % restart if history of xvals smaller TolX';
defopts.TolFun        = 1e-9;    % restart if history of funvals smaller TolFun';
defopts.TolR          = 1e-9;    % restart if step size smaller TolR';
defopts.TolCon        = 1e-9;    % restart if threshold and fitness converge';
```

```
defopts.BoundActive   = 0;       % Flag for existence of bounds
defopts.BoundPenalty  = 0;       % Flag for use of penalty term outside bounds;
defopts.LBounds       = -Inf;    % lower bounds, scalar or Nx1-vector';
defopts.UBounds       = Inf;     % upper bounds, scalar or Nx1-vector';
defopts.bRestart      = 1;       % Flag for restart activation;
defopts.ThreshRank    = 0;       % Flag for threshold based on ranks (Experimental)
defopts.PopMode       = 0;       % Flag for population mode (ToDo);
defopts.Display       = 'off';   % Display 2D landscape while running';
defopts.Plotting      = 'on';    % Plot progress while running';
defopts.VerboseModulo = 1e3;     % >=0, command line messages after every i-th iteration';
defopts.SavingModulo  = 1e2;     % >=0, saving after every i-th iteration';
defopts.bSaving       = 'on';    % [on|off] save data to file';
defopts.bSaveCov      = 0;       % save covariance matrices' ('1' results in huge files);
defopts.funArgs       = [];      % give additional target function arguments
                                          %(scalar, vectors or matrix) if necessary

% Default options for algorithmic parameters
defopts.valP    = 1/exp(1);          % 0.37... Hitting probability
defopts.r       = 1;                 % Step size of the initial covariance
defopts.MaxR    = Inf;               % maximal allowed step size
defopts.MaxCond = 1e6*N;             % maximal allowed condition
defopts.N_mu    = exp(1)*N;          % Mean adaptation weight
defopts.N_C     = (N+1)^2/log(N+1); % Matrix adaptation weight
defopts.N_T     = exp(1)*N;          % Constraint adaptation weight
defopts.inc_T   = 2;                 % Factor for N_T increase at restart (optimization)
defopts.beta    = 1/defopts.N_C;     % Step size increase/decrease factor
defopts.ss      = 1 + defopts.beta*(1-defopts.valP); % Expansion upon success
defopts.sf      = 1 - defopts.beta*(defopts.valP);   % Contraction otherwise

% Option for optimization/design-centering/MCMC sampling mode
% mode = 0 design centering
% mode = 1 optimization
% mode = 2 MCMC sampling

defopts.mode = 1;

% Option for initial threshold
% In the design centering mode c_T is constant. Values smaller than this value are
% considered as feasible points
% In the optimization mode this value is adapted
% In the MCMC mode this threshold is neglected
defopts.c_T = Inf;
```

The design of *gaussAdapt.m* has been inspired by Hansen's CMA-ES implementation. A similar naming scheme and functionality has been adopted. The function *gaussAdapt.m* also contains several child functions (from line 874 onward). Most of them are test functions for optimization, design centering, and sampling. The above listing is almost self-explanatory. All possible GaA settings can be conveniently stored in an options structure and provided to GaA as an argument. The default options have been derived from the present investigations.

## A1.2 Test scripts and support files

All files in the toolbox starting with "testGaA*" contain scripts that show how to initialize and call GaA in the various scenarios.

**Design centering (inopts.mode = 0):**

- testGaAConRegion.m: Shows how GaA approximates the feasible region of a linearly constrained 2D region
- testGaAConElli.m: Shows how GaA approximates the feasible region of an $n$D ellipsoidal region. It also contains details about how to compute the approximate volume of the region.

**Optimization (inopts.mode = 1):**

- testGaASphere.m: Testing GaA on the Sphere function
- testGaARosen.m: Testing GaA on the Rosenbrock function
- testGaAMullerBrown.m: Testing GaA on the Müller-Brown 2D landscape
- testGaADfunnel.m: Testing GaA on the Lunacek's double funnel function
- testGaAKjellstrom2.m: Testing GaA on the Kjellström's function
- testGaARast.m: Testing GaA on the Rastrigin's function
- testGaANoisyS.m: Testing GaA on the noisy sphere function (additive noise)
- testGaACEC2005.m: If the CEC 2005 benchmark suite is available, this script can be used to run GaA on it

**MCMC sampling (inopts.mode = 2):**

- testGaASampler.m: Shows how to sample from several target distributions (Liang's and Haario's test cases)
- testGaANeal.m: Shows how to sample from Neal's funnel distribution (a case where M-GaA should fail!)

**Support files:**

- sphereVol.m: Computes the volume of an $n$D sphere (used for volume computation)
- error_ellipse.m: Plots 2D/3D Gaussian distributions (used for displaying the search trajectory)
- LiangExMat.mat: Data file for one of the target test distributions

## A1.3 GaA in action

We present a design centering and an optimization task.

**testGaAConRegion:** We present an example run of the *testGaAConRegion.m* script, summarized in Fig. 1. The task is design centering. The red region is the feasible region $\mathcal{A}$. GaA adapts a maximum-entropy 2D normal distribution to the region. An average of the final covariance matrices can be used to estimate the volume $Vol(\mathcal{A}) = 39.2857$. For the presented run, GaA estimated a value of 37.4281. The design center is estimated to be at the coordinates (4,4.5).

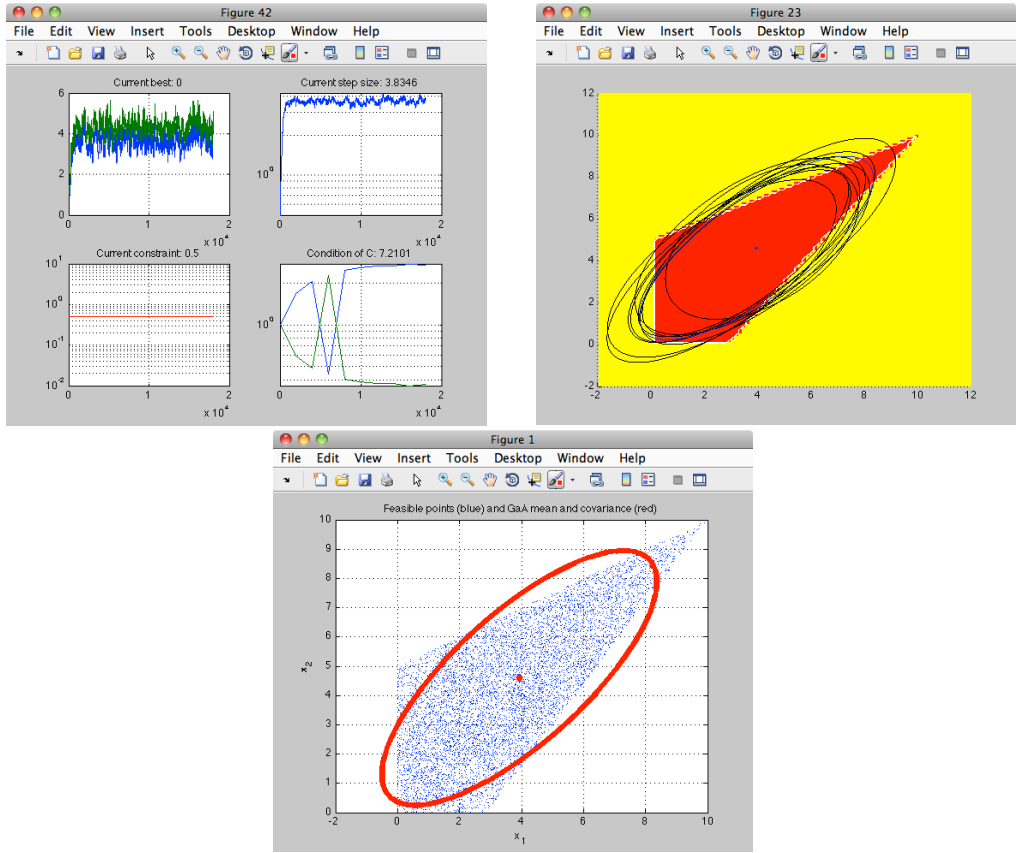



Figure 1: Results of an *testGaAConRegion.m* run.The upper-left MATLAB figure monitors the process of internal variables and the position of the GaA's mean. The upper-right MATLAB figure plots the sequence of covariance matrices of GaA. The lower centered plot depicts the collected feasible samples along with the optimal ellipsoid approximation.

**testGaAMullerBrown:** We present an example run of the *testGaAMullerBrown.m* script, summarized in Fig. 2. The task is optimization of the Müller-Brown surface (Müller and Brown, 1979), a classical 2D double-funnel landscape. In the presented run, GaA performs 3 restarts until the global minimum is found. In the first 3 runs, the competing local minimum is found. The left figure monitors the progress of the best samples found (upper left), the evolution of best fitness and the threshold $c_{\mathrm{T}}(g)$ (lower left), the step size $r^{(g)}$ in the upper right, and the evolution of the condition of the covariance and the length of the eigenvectors (lower right).

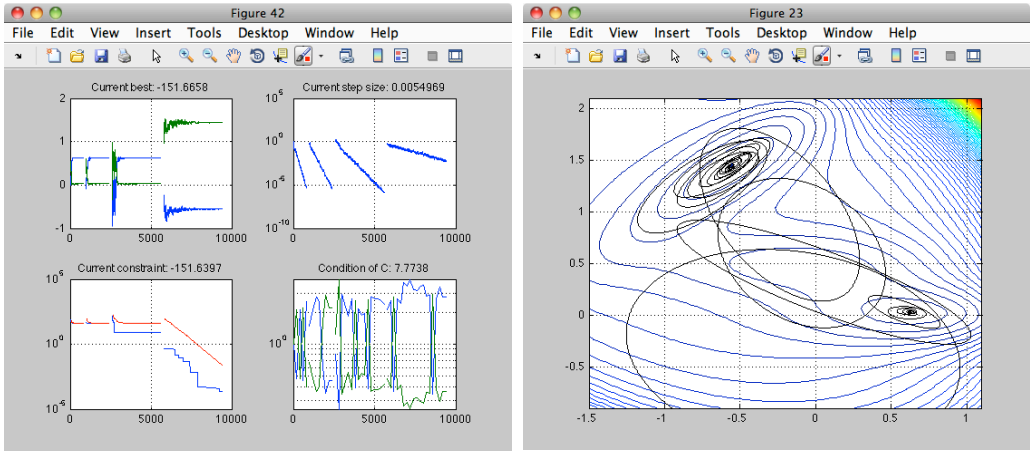



Figure 2: Final result of a *testGaAMullerBrown.m* run. The left MATLAB figure monitors the progress of internal variables and the position of the GaA's mean. The right upper MATLAB figure shows the sequence of covariance matrices of GaA on the Müller-Brown landscape. Four restarts are conducted until the minimum in the top left part of the function is found.

# A2 pCMALib: a parallel MPI-based Fortran 90 library for CMA-ES

## A2.1 Introduction

We present parts of the manual of pCMALib, a parallel Fortran 90 library for the Evolution Strategy with Covariance Matrix Adaptation (CMA-ES). pCMALib is intended as a tool for generic black-box optimization tasks in science and engineering. A schematic overview of pCMALib is given in Fig. 3. The manual is structured as follows: For the impatient user

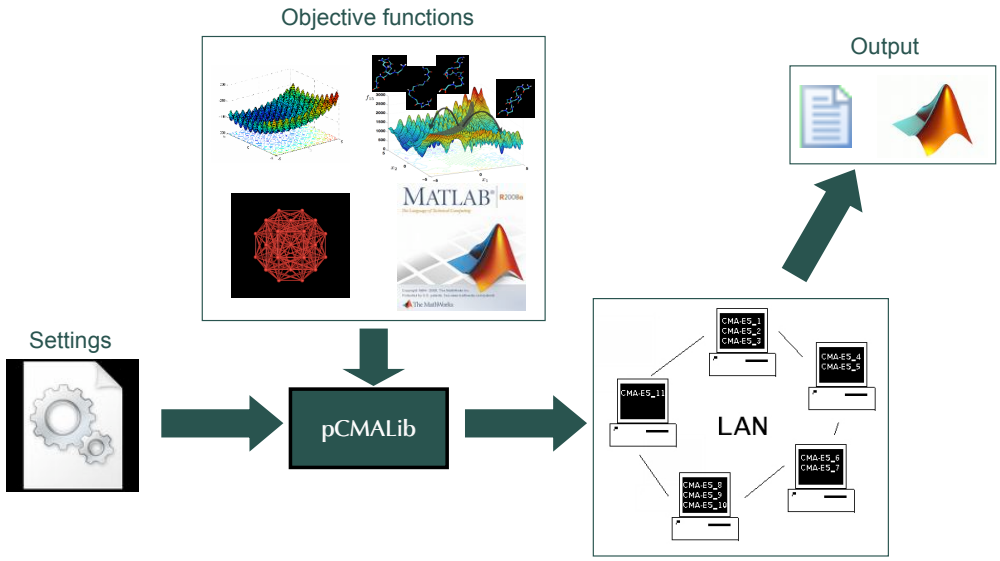



Figure 3: Schematic overview of pCMALib. pCMALib is controlled by a text file that specifies the settings of the algorithmic parameters etc. Within pCMALib, a set of objective functions is available, such as e.g. the CEC 2005 benchmarks test suite and a generic MATLAB test function. pCMALib can run in a single/multi-core or distributed processor environment. Results are stored in either text or MATLAB binary output files.

we first provide a Quick Start section. Section A2.3 summarizes the general library features and the structure of library. The Getting Started Section A2.4 contains a detailed description of the system requirements, the specification of the make.inc/makefile, and the input/output files. We then provide examples of how to test the installation and how to add new objective functions to the library. This Appendix is concluded with a list of known issues with the current version and an outlook on planned developments in pCMALib. We also provide further literature and licensing information.

## A2.2 Quick start

pCMALib is a parallel Fortran 90 library that implements the Evolution Strategy with Covariance Matrix Adaptation (CMA-ES).

The software can be retrieved via our svn repository upon request. If you wish to get started by just typing a few lines and running an example, we provide a quick start section here. You can compile pCMALib without MATLAB and MPI (only LAPACK is required).

1. (unzip pCMALib.zip)
2. (cd pCMALib/)
3. edit make.inc to adapt to the specifications to your environment, leaving MPI and MATLAB variables on their default values
4. `make new` (compiles pCMALib)
5. cd bin/
6. ./libcma ../example_inputs/rastrigin_ipop.txt (run a example)
7. cd rast_ipop
8. ls (check the output data)

If MPI is available you can try the following parallel test case

1. (unzip pCMALib.zip)
2. (cd pCMALib/)
3. edit make.inc to adapt to the specifications to your environment; adapt MPI related flags to your system and set HAS_MPI = 1
4. make clean
5. `make new` (compiles pCMALib)
6. cd bin/
7. mpirun -np 4 ./libcma ../example_inputs/water_pscma.txt (minimize a water cluster)
8. cd water_pscma
9. ls (check the output data, should contain files for each process)

To test the MATLAB interface try the following:

1. (unzip pCMALib.zip)
2. (cd pCMALib/)
3. edit make.inc to adapt the specifications to your environment; adapt MAT related flags to your system and set HAS_MAT = 1; MPI flags should be empty and HAS_MPI = 0
4. make clean
5. `make new` (compiles pCMALib)
6. cd bin/
7. ./libcma ../example_inputs/matlab_test.txt (minimize a water cluster)
8. cd mat_test
9. ls (check the output data, should contain a .mat file)

## A2.3 pCMALib: Features and structure

pCMALib is a parallel Fortran 90 library that implements the Evolution Strategy with Covariance Matrix Adaptation (CMA-ES). We first summarize the key features of pCMALib and then give an introduction to the code design and the library structure.

### A2.3.1 General features

pCMALib includes the following features:

- Optimizing objective functions with standard CMA-ES and IPOP-CMA-ES
- Running embarrassingly parallel CMA-ES instances and PS-CMA-ES in a distributed memory environment with MPI
- Efficient Linear Algebra calculations using LAPACK/BLAS
- Sampling with pseudo-random numbers or low discrepancy sequences
- Coupling CMA-ES with Quasi-Newton (BFGS) methods
- Easy control of a large set of strategy parameters via a single input file
- Interfacing with MATLAB, both for objective function calls and writing binary output files
- Benchmarking on the IEEE CEC 2005 (in Fortran 90) and BBOB (in C) test suites
- Potential energy calculation for Lennard-Jones and TIPnP water clusters (taken from GMIN (Wales et al., 2009))
- Easily extendable to user-provided objective functions

### A2.3.2 Code design and library structure

The core of pCMALib is written in standard Fortran 90. A fundamental goal in the code design is to keep the code structured, easy-to-read, and documented. The algorithmic flow of pCMALib follows Niko Hansen's MATLAB CMA-ES implementation (v. 2.54 from 2006) [1]. The user is able to control all relevant algorithmic parameters via an input text file. Output can be generated in text format or MATLAB binary format. All linear algebra calculations such as matrix multiplications, are done with LAPACK/BLAS routines. pCMALib is able to run several CMA-ES runs in parallel using MPI. There, each CMA-ES instance is a unique MPI process that are mapped onto the available cores. pCMALib includes third-party software, e.g., for generation of quasi-/pseudorandom numbers and for gradient descent. Figure 4 summarizes the library structure. A JAVA-like API will be available soon. We shortly describe the content of all directories and their purpose.

[1] available at http://www.lri.fr/~ hansen/cmaesintro.html

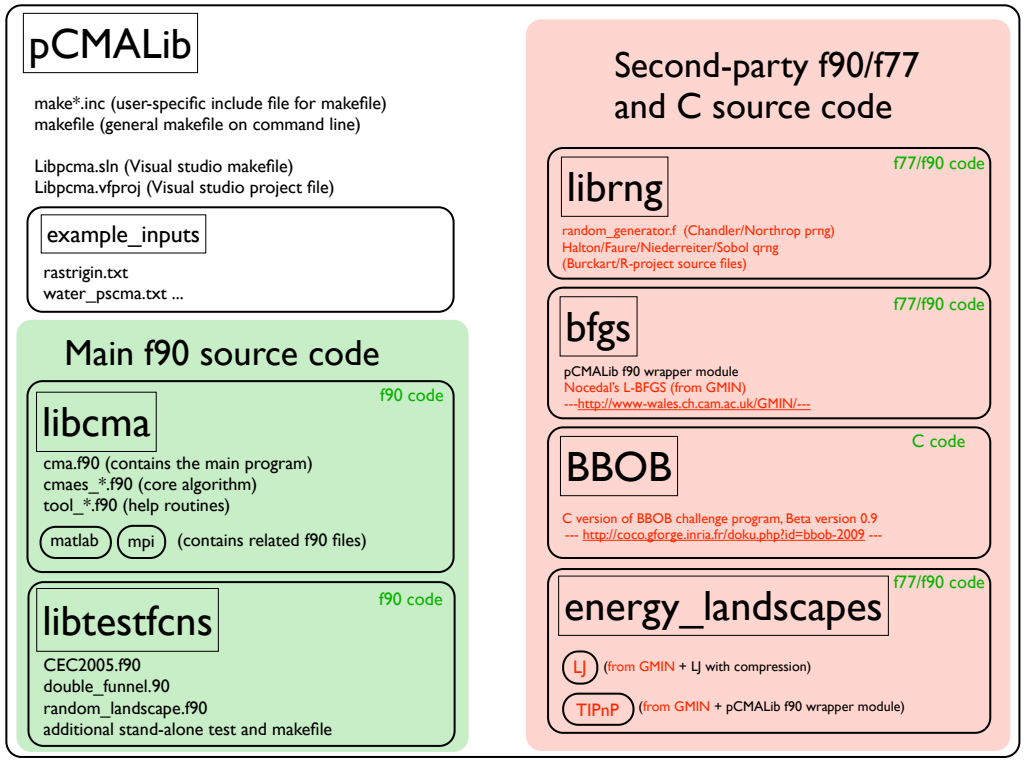



Figure 4: pCMALib library structure

### A2.3.3 General compilation and control files

In the top folder, pCMALib comprises the key compilation files. The files `make*.inc` allow the user to specify all compilation related settings, e.g., the name of the executable, logicals for the use of MATLAB and MPI and their respective location in the system, etc. The `make*.inc` is used by the `makefile` via an include statement. This file serves as input to the `make` command (see A2.4.6 for details). For Windows Visual Studio users we also included `Libpcma.sln`, a Visual studio makefile, and a Visual Studio project file `Libpcma.vfproj`. The directory **example_inputs** contains several text files that serve as example input to pCMALib's executable (see A2.4.7 for details).

### A2.3.4 libcma

The directory **libcma** contains the core f90 source code of pCMALib. Module files that contain key variables and data types needed throughout the source code are named `*_mod.f90`. `cmaes_*.f90` source files contain the core CMA-ES algorithm while `tool_*.f90` files comprise necessary help and support code. The folder **matlab** contains .f90 files that need Fortran

90 libraries provided by MATLAB. The folder **mpi** contains source code that handles all MPI-related operations. The **main program** is included in `cma.f90`.

### A2.3.5 libtestfcns

The directory **libtestfcns** provides several benchmark functions that can be used to test the efficiency of different CMA-ES variants. The `CEC2005.f90` provides a stand-alone Fortran 90 implementation of the IEEE CEC 2005 benchmark test suite (also with make and test files) (Suganthan et al., 2005; Müller et al., 2009b). The `double_funnel.f90` is Lunacek's double funnel test problem (Lunacek et al., 2008). If the user intends to implement further test functions, they should be included in this folder.

### A2.3.6 librng

This directory contains third-party software that is used for random number generation. For pseudo-random numbers we use the Chandler/Northrop f77 implementation of the Marsaglia-Zaman-type subtract-with-borrow generator (`rand_generator.f` ). For low-discrepancy sequences we included several implementations for Halton, Faure, Niederreiter and Sobol sequences (see files for references). All implementations are open source.

### A2.3.7 BBOB

This directory includes the C version of BBOB challenge program, Beta version 0.9, available from `http://coco.gforge.inria.fr/doku.php?id=bbob-2009`. BBOB is the GECCO 2009 benchmark test suite and also provides an excellent test bed for algorithm development and comparison.

### A2.3.8 bfgs

This directory contains Nocedal's ACM TOMS implementation of the Limited Memory BFGS (L-BFGS) algorithm in the version that is included in GMIN (see `http://www-wales.ch.cam.ac.uk/GMIN/` for details). It is wrapped by a pCMALib module.

### A2.3.9 energy_landscapes

The folder **energy_landscapes** currently contains potential energy functions for Lennard-Jones clusters and TIPnP water clusters. With slight modification these files are equivalent to the ones included in GMIN (see `http://www-wales.ch.cam.ac.uk/GMIN/` for details). Users that want to implement and test different cluster energies should include their source files here.

## A2.4 Getting started

### A2.4.1 System requirements

We summarize the necessary and optional system requirements for pCMALib here.

### A2.4.2 Platform

The current implementation of pCMALib has been successfully tested on desktop computers running Windows XP SP2, Windows Vista, Windows 7, Mac OS X 10.4.x, Ubuntu Linux 8.10, and OpenSolaris 2008.11. Calculations on clusters have been successfully conducted on Gentoo Linux 2.6.25 and Redhat Linux CentOS 5.4. Both 32 bit and 64 bit architectures have been tested. However, different combinations of Fortran compilers and MPI distributions have been used on the various machines and platform-independence cannot be guaranteed.

### A2.4.3 Compiler

The **recommended** compiler for pCMALib is the Intel Fortran compiler, version 9.1or higher. Successful compilation with PGI Fortran and gfortran is not guaranteed at the current stage. Before compilation a number of preprocessing statements have to be resolved. We recommend to use the C preprocessor `cpp`. When using the BBOB test suite, a C compiler is also needed. We tested both the GNU C compiler `gcc` and Intel's `icc`.

### A2.4.4 LAPACK/BLAS

Using pCMALib requires a working LAPACK/BLAS installation. We tested both the LAPACK available from `http://www.netlib.org/lapack/` and the one included in the Intel MKL (Math Kernel Library). Benchmark runs revealed that, on Intel processsors, MKL's LAPACK is considerably faster (Müller and Sbalzarini, 2009) and should be used.

### A2.4.5 MPI

Running pCMALib in parallel setting requires the installation of an MPI library. We tested and recommend both the OpenMPI (1.2.6, 1.2.8, 1.3,1.4, and 1.5) and the Intel MPI library. Again, running pCMALib with Intel MPI jointly with Intel MKl on Intel cores gives superior performance (Müller et al., 2009a).

### A2.4.6 Compiling pCMALib

Before compilation, some system-specific configurations have to be set. This is done by adapting the variables in one of the provided `make*.inc` files.

Listing A.1: Snippet of the make.inc file

```
############################################################
# PCMALIB make include file
# December 2009
#
```

```
# Georg Ofenbeck,
# MOSAIC Group, ETH Zurich, Switzerland
#####################################################################

# Name of the program
LIB_CMA := libpcma.a

########## SHELL OPTIONS #################

SHELL    := /bin/sh

######## OUTPUT Folder #################
# will create directories BUILDDIR/bin, BUILDDIR/objects and BUILDDIR/include here
BUILDDIR:= .
...
```

We have a specific `make_brutus.inc` for the ETH Zurich cluster Brutus and a user-specific `make.inc`. There, the user can specify all relevant variables, e.g., the name of the executable by `LIB_CMA := libpcma.a`, the directory where to build the object and binary files, the Fortran 90 compiler, the locations of the LAPACK/MPI/MATLAB installations, etc.

Listing A.2: Beginning of the makefile

```
#-----------------------------------------------------------------------
# pCMALib: a parallel fortran 90 library for the evolution strategy with
#          covariance matrix adaptation
# Christian L. Mueller, Benedikt Baumgartner, Georg Ofenbeck
# MOSAIC group, ETH Zurich, Switzerland
#-----------------------------------------------------------------------
include make.inc

# Default value for the dependency pre-processor
# = same as code-generating pre-processor
DEPCPP ?= $(CPP)
$(CPP) = cpp
...
```

The makefile first resolves all source dependencies and preprocessor statements. Then, the resulting sources are compiled. The makefile contains four targets:

Listing A.3: makefile targets

```
.DEFAULT: ;
# Default target is everything
all: $(TARGET)

...

# at the install part we copy the input files and the programm itself to
install:
        $(shell mkdir $(INSTALL_DIR))
        $(shell mv $(TARGET) $(INSTALL_DIR))
        $(shell cp -r $(CEC2005_DIR)/supportData $(INSTALL_DIR))
```

```
# Get rid of the .d's too. Notice this may produce weird behavior during
# the next invocation, since the makefile itself needs the .d's.
clean:
        rm -f $(OBJECTS)
        rm -f $(MODULES)
        rm -f $(TARGET)
        rm -f $(DEPENDENCIES)

# Make new
new: clean all install
```

Execute `make new` in the main folder. The makefile should generate a **objects** and a **bin** folder. In the **bin** folder you will find the executable with the specified name, e.g., libpcma.a. Depending on your specification this file can be run on single or multiple cores using `(o)mpirun`. The program is controlled by a single text file as outlined in the next section.

### A2.4.7 Controlling pCMALib: the program input file

pCMALib's program is controlled by a input file that specifies all algorithmic settings. The **example_inputs** folder contains several examples. We show here the example used in the Quick start section where IPOP-CMA-ES is run on the shifted 10D Rastrigin function (Function f9 from the CEC 2005 benchmark test suite (Suganthan et al., 2005)).

Listing A.4: Example input file

```
#-------------------------------------------------------------------------
# pCMALib: a parallel fortran 90 library for the evolution strategy with
#          covariance matrix adaptation
# Christian L. Mueller, Benedikt Baumgartner, Georg Ofenbeck
# MOSAIC group, ETH Zurich, Switzerland
#-------------------------------------------------------------------------

# output data is saved into this folder (relative to workdir)
OUTPUT_FOLDER = rast_ipop
# which function of the CEC2005 or BBOB benchmark suite to use
BENCHFCTNR = 9
# Dimension of the problem
DIMENSIONS = 10
# Upper bounds on all dimensions
ALLDIM_UBOUNDS = 5
# Lower bounds on all dimensions
ALLDIM_LBOUNDS = -5
#the global optimum
GLOBAL_MIN = -330
# use the CEC2005 benchmark suite as target function
USE_CEC = true
# usage of Quasi Random Sampling
QR_SAMPLING = true
#  this only works with Sobol R implementation! (0) no scrambling
# (1)owen type scrambling, (2)faure-tezuka type scrambling,
# (3)owen,faure-tezuka type scrambling
QR_SCRAMBLING = 0
# (0)Moros Inverse, (1)Peter J. Acklam.s Inverter, (2)Inverter from R
QR_INVERTER = 1
#  (0)Sobol, (1)Sobol R implementation, (2)Halton, (3)Halton R
#  implementation, (4)Faure (buggy!), (5)Niederreiter
QR_SAMPLER = 1
# Successful run if global_min -f(x) < accuracy
ACCURACY = 1.E-8
# if multi restart CMA (IPOP) should be used
RESTART_CMA = true
# (0) restart randomly within bounds, (1) restart from point of
# convergence, (2) restart from same startpoint all the time
RESTART_TYPE = 1
# factor by which the population size is increased every restart
INCPOPSIZE = 1.3
#the folder where to find the supportData folder
CECFOLDERS = ./
```

It is important that the input parameters are spelled correctly. Wrongly spelled input parameter names are ignored.

As in the MATLAB version of CMA-ES, there are many options that can be set. The following tables summarize all available options. In the code, most of the default settings are defined in `libcma/cmaes_opts_mod.f90`.

| Name | Default Value | Explanation |
|---|---|---|
| **Stop Criteria** | | |
| STOPFITNESS | -Inf | Stop if $f(x) <$ StopFitness |
| STOPMAXFUNEVALS | Inf | Maximal number of FES |
| STOPMAXITER | $\frac{1e3\,(n+5)^2}{\sqrt{\text{PopSize}}}$ | Maximal number of iterations |
| STOPTOLX | $1e\text{-}11\,max(\sigma_{initial})$ | Stop if x-change $<$ StopTolX |
| STOPTOLUPX | $1e3\,max(\sigma_{initial})$ | Stop if x-change $>$ StopTolUpX |
| STOPTOLFUN | $1e$-12 | Stop if fun-change $<$ StopTolFun |
| STOPTOLHISTFUN | $1e$-13 | Stop if back fun-change $<$ StopTolFun |
| STOPTIME | true | usage of a stop time |
| STOPTIMEHH | 0 | stop after given hours |
| STOPTIMEMM | 0 | stop after given minutes |
| STOPTIMESS | 0 | stop after given seconds |
| STOPONWARNINGS | true | stop if any of the warnings occur |
| **CMA-ES Strat. Params.** | | |
| ABS_SIGMA | 0 | size of the inital $\sigma$ as absolut value |
| REL_SIGMA | 0.2 | size of the inital $\sigma$ relative to the size of the box constraints - only used if ABS_SIGMA is not set |
| POPSIZE | $4 + \lfloor 3 \ln(n) \rfloor$ | Population size $\lambda$ |
| PARENTNUMBER | $\lfloor \frac{\lambda}{2} \rfloor$ | Parent number $\mu$ |
| RECOMBINATIONWEIGHTS | 3 | Super–linear (3), linear (2) or equal (1) |
| **PS-CMA-ES Strat. Params.** | | |
| PSCMA | false | Switch PS-CMA-ES on or off |
| PSOWEIGHT | 0.7 | weights between PSO-based and local covariance matrix (equivalent to $c_{\text{p}}$ in eq. (4.16)) |
| PSOFREQ | 200 | Interval length $I_{\text{c}}$ between PSO updates (see Section 4.3.1) |
| **Sampling Options** | | |
| QR_SAMPLING | true | usage of Quasi Random sampling |
| QR_SAMPLER | 1 | (0)Sobol, (1)the Sobol R implementation<br>(2)Halton<br>(3)Halton R implementation<br>(4)Faure (buggy!)<br>(5)Niederreiter |
| QR_SCRAMBLING | 0 | this only works with Sobol R implementation!<br>(0)no scrambling<br>(1)owen type scrambling<br>(2)faure-tezuka type scrambling<br>(3)owen,faure-tezuka type scrambling |
| QR_INVERTER | 1 | (0)Moros Inverse<br>(1)Peter J. Acklam's Inverter<br>(2)Inverter from the R Implementation |

Table 1: Available pCMALib options (1)

| **Restart Settings** | | |
|---|---|---|
| RESTART_CMA | false | Flag if restart CMA (IPOP) should be used |
| RESTART_TYPE | 0 | (0) restart randomly within bounds<br>(1) restart from point of convergence<br>(2) restart from same startpoint all the time |
| RESTARTS | 0 | limit on how many restarts are allowed, 0 = unlimited |
| INCPOPSIZE | 1.25 | factor by which the population size is increased every restart |
| MAXINCFAC | 100 | the maximum fold increase of the population compared to the initial population size |
| **Benchmark Opts.** | | |
| BENCHMARK | false | Switch benchmark on or off, this causes pCMALib to record and keep track of several variables that are required in the CEC 2005 benchmark protocols |
| RECORD_ACCURACY | 0.0 | record when the CMA-ES reaches this level of accuracy |
| RECORD_BESTHIST | false | record a history of the fitness over time |
| RECORD_MODULO | 100 | $MAX\_FES/RECORD\_MODULO$ gives the number of records |
| **Others** | | |
| DIMENSIONS | | Dimension $n$ of the problem |
| GLOBAL_MIN | 0.0 | Global minimum (if available) |
| ACCURACY | 0.0 | Successful run if $\|\texttt{GLOBAL\_MIN} - \texttt{f(x)}\| \leq \texttt{accuracy}$ |
| EVALINITIALX | true | Evaluate initial solution |
| WARNONEQUALFUNCTIONVALUES | true | Report warning if all function values in a generation are identical |
| FLGGENDATA | true | Flag if complete output data should be generated (can result in huge files!) |
| INTGENDATA | 1 | Integer interval to log output data |
| FLGOUTTXT | true if compiled without matlab, otherwise false | saves the results of the pCMAlib run into textfiles |
| FLGGENTRACE | false | saves only trace of best solutions and their fitness values into textfiles |
| FUNCNAME | ' ' | Name of the function that will be reported in output |
| VERBOSEMODULO | 100 | Messaging after every i-th iteration in the console |
| OUTPUT_FOLDER | 'out' | output data is saved into this folder (relative to workdir) |
| SILENTMPI | true | flag if only process with rank 0 should report output in the console |
| USE_SEED | false | if a seed for the RNG should be used |
| SEED_FOLDER | 'false' | folder containing the seed file 'seed.txt' |

Table 2: Available pCMALib options (2)

| **Boundary setting** | | |
|---|---|---|
| ALLDIM_LBOUNDS | -Inf | Lower bounds on all dimensions |
| ALLDIM_UBOUNDS | Inf | Upper bounds on all dimensions |
| USE_INIT_BOUNDS | false | if special bounds should be used for initalization of population |
| INIT_UBOUNDS | Inf | Upper bounds for initialization |
| INIT_LBOUNDS | -Inf | Lower bounds for initialization |
| **Objective functions** | | |
| USE_LJ_COMP | false | use the Lennard Jones potential with compression as target function |
| USE_DF | false | use the DoubleFunnel benchmark as target function |
| USE_MATFUNC | false | use the template for a Matlab target as function |
| USE_RANDOM_LANDSCAPE | false | use the random landscape test as target function |
| USE_CEC | false | use the CEC2005 benchmark suite as target function |
| USE_BBOB | false | use the BBOB benchmark suite as target function |
| USE_LJ | false | use the Lennard Jones potential as target function |
| USE_TIP | false | use the TIP4P water potential as target function |

Table 3: Available pCMALib options (3)

### A2.4.8 Output files

There are several modes of output generation in pCMALib, which are controlled by the four input parameters FLGGENDATA, INTGENDATA, FLGOUTTXT, and FLGGENTRACE. The files are generated in the folder specified by the input parameter OUTPUT FOLDER.

If FLGGENDATA is true, pCMALib will write out every INTGENDATA **generation** almost all CMA-ES data, such as the populations, the covariance matrices, the Cholesky matrices, the evolution path, etc. Depending on the dimensionality, the run length, and the write out interval length, this may result in **HUGE** text files. When doing production runs with pCMALib, the user is advised to use the settings with caution. If MATLAB is available, in addition to the text file, a binary MATLAB file cmaesData.mat is generated that includes the summary data given below, but **NOT** the traces along the optimization run.

If FLGGENDATA is false, the user has the possibility to just write the summary data (see below) in text files by setting FLGOUTTXT true. MATLAB is not required for this.

The list of summary output files is provided in Table 5. If MATLAB is available, these data are stored in a structured binary MATLAB .mat file. It follows the standard output from the Hansen's CMA-ES MATLAB version, plus some pCMALib-specific output. Finally, if the user is only interested in the trace of current best candidate solutions and its corresponding fitness values, these can be generated by setting the FLGGENTRACE true. Then, the text files

| **Add. objective function settings** | | |
|---|---|---|
| WRITE_PDB | false | if a pdb file representing the current best solution is written in intervals according to the VerboseModulo setting |
| LJ_COMP | 1 | compression parameter $\mu_{\text{comp}}$ for the Lennard Jones potential with compression |
| DF_S | 0 | setting for parameter 's' for the Double Funnel benchmark |
| DF_RAST | true | if rastrigin function should be applied to the Double Funnel benchmark |
| CECFOLDERS | " | where to find the supportData folder for the CEC2005 benchmark suite relative to the working directory |
| BENCHFCTNR | 1 | which function of the CEC2005 or BBOB benchmark suite to use |
| **BFGS Settings** | | |
| BFGS_USE | false | if BFGS should be used to assist CMA. This is still in development! |
| BFGS_FACTR | | not used at the moment |
| BFGS_PGTOL | | not used at the moment |
| BFGS_GRAD_STEPSIZE | | step size used for the gradient approximation |
| BFGS_POSITION | 2 | 1 = replace X values by local minimum X<br>2 = replace F values with F values at local minimum |
| BFGS_CENTRAL_DIFFERENCE | false | if central difference should be used, otherwise backward difference is utilized |
| BFGS_DGUESS | | the guess for the inital step size for the line search |
| BFGS_STPMAX | | upper bounds for the step in the line search |
| BFGS_STPMIN | | specify lower for the step in the line search |
| BFGS_GTOL | | controls the accuracy of the line search routine MCSRCH |

Table 4: Available pCMALib options (4)

| | |
|---|---|
| out_bestever_f.txt | the best fitness value found during the optimization |
| out_bestever_x.txt | the best input vector x found during the optimization |
| out_bestever_evals.txt | the evaluation where the best fitness value has been found |
| out_countEvalNaN.txt | the number of invalid samples drawn during the optimization run |
| out_countEval.txt | the number of samples drawn during the optimization |
| out_countIter.txt | the number of CMA Iteration done during the optimization |
| out_countOutOfBounds.txt | the number of samples that fell outside of the Bounds given to CMA |
| out_funcName.txt | the name of the function like given in the input file.<br>If a benchmark is utilized, it returns the name of the function evaluated |
| out_insigma.txt | the initial sigma CMA starts with in absolute numbers |
| out_lambda.txt | the initial lambda CMA starts with |
| out_mueff.txt | effective population weight |
| out_mu.txt | number of selected individuals per generation |
| out_N.txt | dimension of the problem |
| out_settings.txt | summary of all used settings |
| out_stopflag.txt | info about applied stopping criteria |
| out_weights.txt | the weights used for the ranking |
| out_xstart.txt | the initial x_mean of CMA-ES - not used at the moment |
| seed.txt | the initial random number seed - for reproducing identical runs |

Table 5: Summary output files from pCMALib

besteverF.txt and besteverX.txt contain these values at every INTGENDATA **generation**.

If pCMALib is run in MPI mode, the output files are generated separately by each process. The resulting file names are the same as in the single-process case, but with the extension $\_RANK$, i.e., the cmaesData.mat for the process with rank 0 is called $cmaesData\_0.mat$.

For the specific case where LJ or TIPnP water clusters are optimized, the user has the possibility to write out PDB files with the LJ or water atom coordinates. For this purpose, the flag $WRITE\_PDB$ should be set true. The write frequency is controlled by the input parameter VERBOSEMODULO.

## A2.5 Test example

### A2.5.1 IPOP-CMA-ES

- Unzip / checkout
- Change all parameters in the make.inc file. MPI and MATLAB Include/Library paths are not required for this example. Make sure that HAS_MPI, HAS_MAT, and BBOB are all set to 0.
- in the main folder, execute `make new`
- cd to the newly created **bin** folder

- execute `./libpcma.a ../example_inputs/rastrigin_ipop.txt`
- you should get output on the console similar to the one listed below

Listing A.5: Console output generated by pCMAlib for the Rastrigin IPOP example

```
   ****************************************************
Warnings:
n= 10: ( 5 , 10 )-CMA-ES on function Shifted Rastrigins Function
        Iterat ,      #Fevals:      Function Value
           1 ,            12 :   -134.484715127431
         100 ,          1002 :   -319.957567812663
         200 ,          2002 :   -320.050414465281
         262 ,          2622 :   -320.050414466886
         262 ,          2622 :   -320.050414466886
------- Restart #            1  Reason: warnequalfunvals
n= 10: ( 6 , 13 )-CMA-ES on function Shifted Rastrigins Function
-----------------------------------
Initial sigma    2.00000000000000
-----------------------------------
        Iterat ,      #Fevals:      Function Value
         300 ,          3116 :   -260.736124312701
         400 ,          4416 :   -326.020162842026
         473 ,          5365 :   -326.020163771627
         473 ,          5365 :   -326.020163771627
------- Restart #            2  Reason: warnequalfunvals

.
.
.

------- Restart #            2  Reason: warnequalfunvals

n= 10: ( 45 , 91 )-CMA-ES on function Shifted Rastrigins Function
-----------------------------------
Initial sigma    2.00000000000000
-----------------------------------
        Iterat ,      #Fevals:      Function Value
        7000 ,        524992 :   -281.725617082436
        7071 ,        531453 :   -329.999999995965

GLOBAL Bestever.f:   -329.999999995965
GLOBAL Bestever.x:
      1.900E+00
     -1.564E+00
     -9.788E-01
     -2.254E+00
      2.499E+00
     -3.285E+00
      9.759E-01
     -3.666E+00
      9.850E-02
     -3.246E+00
Stopflag:fitness
```

- *ls rast_test* should yield a list of files generated by the optimization run as shown in the previous section.

### A2.5.2 PS-CMA-ES

- Unzip / checkout
- Change parameters in the make.inc file. MPI paths are required for this example. Make sure that HAS_MAT and BBOB are set to 0, while HAS_MPI is set to 1
- in the main folder execute `make new`
- cd to the newly created **bin** folder
- execute `mpirun -n 4 ./libcma ../example_inputs/water_pscma.txt`. This command structure might vary depending on your installed MPI implementation.
- you should get output on the console similar to the one listed below

Listing A.6: Console output generated by pCMAlib for the parallel Rastrigin IPOP example

```
*****************************************************
Warnings:
*****************************************************
Started MPI-CMA
To guarantee a decent console output only Process 0 is shown
 All output data is saved to
folder water_pscma


*****************************************************
-----------------------------------
PSO configuration:
Weight:    0.800000000000000
Frequency:          1000
-----------------------------------
Initial sigma    2.40000000000000
-----------------------------------
n= 48: ( 7 , 15 )-CMA-ES on function
        Iterat ,      #Fevals:     Function Value
           1 ,            17 :   -24.7688838771480
         100 ,          1502 :   -91.9960387430328
         200 ,          3002 :   -141.671714969162
         300 ,          4502 :   -149.296251112378
         400 ,          6002 :   -194.206246961448
         500 ,          7502 :   -203.078431185465
         600 ,          9002 :   -207.834413560923
         700 ,         10502 :   -208.866055139447
         800 ,         12002 :   -209.640611386233
         900 ,         13502 :   -210.402527129655
        1000 ,         15002 :   -211.520111342927
        1100 ,         16502 :   -212.422564260838
        1200 ,         18002 :   -212.949637423201
        1300 ,         19502 :   -213.668174681497
```

```
         1400  ,        21002  :   -226.061157943448
         1500  ,        22502  :   -237.363561233379
         1600  ,        24002  :   -239.168766472368
.
.
.
 GLOBAL Bestever.f:   -243.051022815494
GLOBAL Bestever.x:
      1.423E+00
   1.402E+00
  -2.535E+00
  -2.564E-01
  -6.437E-01
  -3.328E+00
  -8.623E-01
  -9.006E-01
...
  -1.536E+00
  -9.409E-01
  -2.305E-01
  -3.482E+00
  -6.000E+00
   9.542E-01
   4.354E+00
   1.263E+00
  -2.616E+00
  -1.269E+00
  -1.310E+00
Stopflag:tolfun
```

- *ls water_pscma* should yield a list of files generated by the optimization run, as described in the previous section, but this time one file for each process should exist of the form FILENAME_{Rank.txt}

## A2.6 Adding new objective functions

Adding user-defined objective functions is rather simple. In the user folder there is a template that you can adapt for your needs. If the name of the template function is changed, you also have to rename `EXTERNAL user_function` in `cmaes.f90` and its call further down in the main as well. Further test functions can also be found in either **testfcns** or **energy_landscapes** folders. They also have to be declared as `EXTERNAL user_function` in `cmaes.f90` in order to work properly.

Listing A.7: Snippet of the main.f90

```
!-----------------------------------------------------------------------
!  Routine        :              user_function
!-----------------------------------------------------------------------
!
!  Purpose        :
!
!  Remark         : at the moment pcmalib will only call single values
!                   at once meaning it will send 'vars' of size m x 1 and
```

```
!                   expect a vector of size n=1 back. This might change
!                   in the future and therefore it is recommended
!                   to write the function in a way that it can handle
!                   multiple input vectors in form
!                   of a matrix and return a vector of results
!
!  Input        :
!                  m        (I) m Dimension of the matrix (Rows)
!                                = Dimension of Vectors
!                  n        (I) n Dimension of the matrix (Columns)
!                                = # of Vectors to process
!  Input(optional): lbounds / ubounds - m dimensional array
!                                 with boundaries given to CMA-ES
!
!
!  Input/Output:  vars     (R) the matrix with the input values of size m*n
!
!  Output:        res      (R) the vector with the results of size n
!
!-------------------------------------------------------------------------------


SUBROUTINE user_function(res,vars,m,n,lbounds,ubounds)
USE cmaes_param_mod
!-------------------------------------------------------------------------------
!  Parameters
!-------------------------------------------------------------------------------
REAL(MK),DIMENSION(n),INTENT(out)                 :: res
REAL(MK),DIMENSION(m,n),INTENT(in)                :: vars
INTEGER,INTENT(in)                                :: m
INTEGER,INTENT(in)                                :: n
REAL(MK),DIMENSION(m),OPTIONAL                    :: lbounds
REAL(MK),DIMENSION(m),OPTIONAL                    :: ubounds


!your code here!!!!!
WRITE(*,*) 'no code provided in the user_function yet'
STOP


END SUBROUTINE user_function
```

As can be seen from the current template for objective functions it is not possible to explicitly provide additional parameters to the objective functions. If the user function needs additional parameters, they have to be provided via global/MODULE variables. An example of how do that can be found in the double funnel test function case. This objective function has a parameter $s$ that tunes the size and depth ratio of two funnels. It is provided through the input parameter $DF\_S$ in the input file.

## A2.7 Known issues

- Scrambling caused trouble returning 0 values for all samples on the ETH cluster after some million iterations. This could not be reproduced on a desktop machine.

- IPOP (restart) settings can cause trouble when not limited by the MAXINCFAC control parameter. This is due to the exponential increase in population size and hence the linked exponential increase in memory requirements.

- The matlab engine is not the most reliable – it happened sometimes that the program just freezes during the writing of the .mat file.

- Very long path names (>200 characters) are cut off and might cause problems.

- If FLGGENDATA is set to true and INTGENDATA small, the output files can become very big due to the fact that they are written as text files.

- After make.inc has been altered, manual cleaning of the previous make run may be required (by manually deleting the **objects** folder).

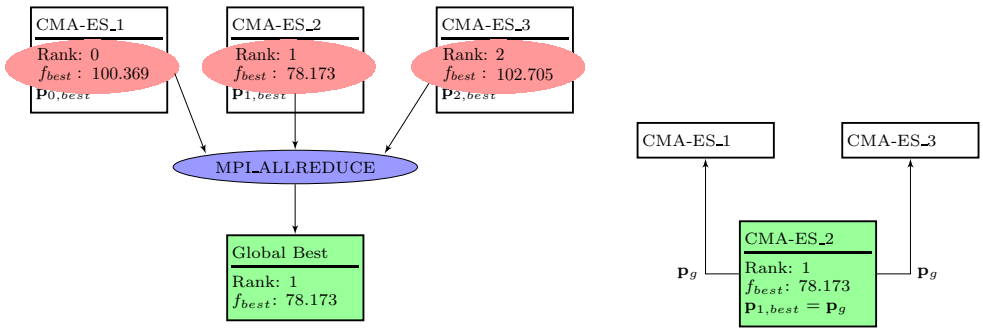
## A2.8 MPI structure in PS-CMA-ES

### A2.8.1 MPI protocol for PS-CMA-ES

The parallel CMA-ES code is implemented using on MPI. Multiple CMA-ES instances need to communicate in order to integrate parallel information. Each CMA-ES instance is a computational process with separate address space. These processes are assigned to to multiple processors (for example different PCs in a Local Area Network (LAN)). If such hardware is not available, MPI allows to execute parallel code on single processors.

### A2.8.2 GLOBAL_BEST-Communication

In MPI, each process is assigned a unique number: the process rank. Multiple processes can be distinguished by their rank. InPS-CMA-ES, each CMA-ES instance has to inform the other swarm members about its current best candidate solution. Fig. 5 illustrates our procedure. A 2-dimensional array, called F_BEST, is introduced. It stores the current best fitness value, as well as the process rank (illustrated by light red ellipses in Fig. 5). The MPI collective communication routine *MPI_ALLREDUCE* is used to find the global best function value within all F_BEST arrays. Since the array also contains the process rank, the corresponding CMA-ES instance is known. In a second step (Fig. 5), this process broadcasts the position of its current optimal solution to the other swarm members.
In order to reduce communication overhead, the broadcast is only performed, if the current optimum has changed.




(a) MPI Allreduce (b) MPI Broadcast

Figure 5: MPI communication of the global best position $\mathbf{p}_g$.

### A2.8.3 Excluding processes from communication

If a CMA-ES instance has converged and stopped its search, it needs to be excluded from communication. There are two possible scenarios, how a safe program termination can be ensured, and communication deadlocks can be avoided: One is that the process waits until all other running processes have reached the same state, such that all processes can be finalized synchronously. Such an approach, however, seems inefficient since resources are kept allocated until the last process has converged.
Therefore, we favor a more dynamic approach: Whenever a process stops, communication is adapted, such that this specific process is excluded. To describe our method, definitions of MPI groups and communicators have to be given:

**Definition A.1.** *A **group** is an ordered set of processes. Each process in a group is associated with a unique integer rank. Rank values start at zero and go to N-1, where N is the number of processes in the group. A group can be associated with a communicator object.*

**Definition A.2.** *A **communicator** encompasses a group of processes that may communicate with each other. All MPI messages must specify a communicator. The communicator that comprises all tasks is MPI_COMM_WORLD.*

Using these MPI objects, we adjust the communication as follows: At the end of each CMA-ES generation we check whether one or more processes have met a stopping criterion and are about to terminate. If this is the case, the following rules are applied:

1. Ranks of terminating processes are collected.
2. *MPI_COMM_GROUP()* and *MPI_GROUP_EXCL()* are used to build a new communication group that excludes the terminating ranks.
3. A new rank for each process is assigned in the group (*MPI_GROUP_RANK()*).
4. A group communicator using *MPI_COMM_CREATE()* is created.

5. Calculations are continued using the newly created communicator. All processes that are not in the scope of the communicator are terminated.

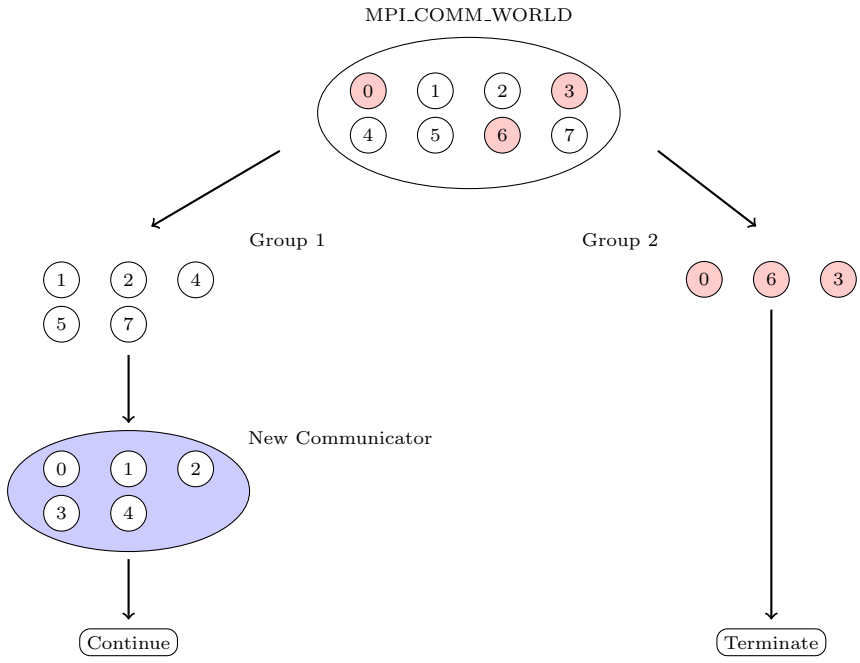



Figure 6: Dynamically excluding converged processes from communication

Figure 6 exemplifies this MPI group and communicator management. At the very top of the figure, all processes are within the same communicator environment (MPI_COMM_WORLD). Colored in light red are processes that will stop. Following the chart, two groups are formed. One group contains running processes (Group 1), the other contains stopping processes (Group 2)[1]. Based on Group 1, a new communicator is created. Note, that the processes are assigned a new rank starting again from 0.

## A2.9 Benchmarks

We benchmark the computational performance and parallel efficiency of pCMALib on multi-core and distributed-memory computers.

## A2.10 Multi-core shared memory

We first test the on-chip performance of the library on an Apple MacPRO with 2 dual-core 3 GHz Intel Xeon processors, a 4 MB L2-cache per processor, and $8 \times 1$ GB of RAM. The

[1]Group 2 is not a valid communication group. Instead, it contains processes with rank value *MPI_UNDEFINED*

library was compiled with the Intel Fortran compiler version 9.1 and optimization level O3, and linked against OpenMPI version 1.2.6.

We follow the CEC 2005 test suite protocol to assess the computational efficiency of our implementation. Three time measures are defined in the protocol: $T_0$ is the CPU time for 1 000 000 standard mathematical operations, $T_1$ is the time needed to evaluate function $f_3$ – a shifted, rotated, highly conditioned elliptic function – 200 000 times in dimensions $n = 10, 30, 50$, and $\hat{T}_2$ is the mean time over five executions of the complete algorithm with 200 000 evaluations of function $f_3$ each. The computational cost of the algorithm is quantified by the ratio $(\hat{T}_2 - T_1)/T_0$.

We benchmark our implementations of the standard CMA-ES (on a single core), the parallel CMA-ES, and the parallel PS-CMA-ES. The standard CMA-ES is run with the standard strategy parameter settings (Hansen, 2008) on a single core. The parallel CMA-ES benchmark uses 4 independent CMA-ES instances on the available 4 cores of the computer, without any communication between the instances. PS-CMA-ES is run with a swarm size of $S = 4$ on the 4 cores of the computer and the communication interval is set to the standard value of $I_c = 200$ (Müller et al., 2009b). The system configuration and CPU time measurements are summarized in Table 6, following the CEC 2005 test suite requirements (Suganthan et al., 2005). For $n = 10$ and 30 we observe that the $\hat{T}_2$ of the three methods are comparable. For $n = 50$, the computational cost of PS-CMA-ES dominates due to the complexity of the $n$-dimensional matrix rotations. For comparison, we cite measurements of $\hat{T}_2$ for LR-CMA-ES

| **System** | Mac OS X 10.4.11 |
|---|---|
| **CPU** | 2× Dual-Core Intel Xeon 3.00GHz |
| **RAM** | 1GB |
| **Language** | Fortran 90 |

| **CMA-ES** | $T0$ | $T1$ | $\hat{T}2$ | $(\hat{T}2 - T1)/T0$ |
|---|---|---|---|---|
| $n = 10$ | 9.53e-2 | 3.02e-1 | 2.71e+0 | 2.53e+1 |
| $n = 30$ | | 2.26e+0 | 1.13e+1 | 9.49e+1 |
| $n = 50$ | | 6.49e+0 | 3.04e+1 | 2.51e+2 |
| **Parallel CMA-ES** | | | | |
| $n = 10$ | | 3.02e-1 | 3.96e+0 | 3.84e+1 |
| $n = 30$ | | 2.26e+0 | 1.39e+1 | 1.22e+2 |
| $n = 50$ | | 6.49e+0 | 3.53e+1 | 3.02e+2 |
| **PS-CMA-ES** | | | | |
| $n = 10$ | | 3.02e-1 | 3.87e+0 | 3.75e+1 |
| $n = 30$ | | 2.26e+0 | 1.55e+1 | 1.39e+2 |
| $n = 50$ | | 6.49e+0 | 5.04e+1 | 4.61e+2 |

Table 6: System configuration and measured CPU times in seconds for standard CMA-ES, parallel CMA-ES, and parallel PS-CMA-ES.

and IPOP-CMA-ES determined by Auger and Hansen (Auger and Hansen, 2005b,a) using MATLAB 7.0.1 on Red Hat Linux 2.4 running on a 3 GHz Intel Pentium 4 processor with 1 GB RAM. For $n = 10, 30, 50$, LR-CMA-ES took $\hat{T}_2 =$ 51s, 45s, 68s, and IPOP-CMA-ES $\hat{T}_2$ = 17s, 24s, 56s, respectively.

### A2.10.1 Distributed memory

We assess the parallel efficiency of our implementations of CMA-ES and PS-CMA-ES on a distributed-memory computer cluster on the constrained random fitness landscape $F_{\mathrm{rand}}(\mathbf{x}) = Y$, where $\mathbf{x}$ is defined in the bounded subset $[-100, 100]^n \in \mathbb{R}^n$. For any $\mathbf{x}$, $Y$ is drawn from the uniform distribution $\mathcal{U}(-100, 100)$. Each algorithm evaluates the fitness function 500 000 times (corresponding to drawing 500 000 uniformly distributed random numbers) on $N_{\mathrm{proc}} = 1, \ldots, 64$ processor cores. The number of CMA-ES instances – or the swarm size in PS-CMA-ES – is always chosen equal to $N_{\mathrm{proc}}$ in order to avoid cache and memory congestion effects. Distributing a problem of fixed size onto an increasing number of processors measures the **strong** scaling of the algorithms, where the workload per processor decreases and the communication overhead increases. The random landscape $F_{\mathrm{rand}}$ ensures several properties that are indispensable for an unbiased assessment of the parallel scaling. First, the computational cost of evaluating the objective function is independent of the search dimension and the specific optimization path. Second, the random landscape guarantees that all CMA-ES instances experience the same search space. We perform three benchmarks with varying values of the strategy parameter $I_c$ in order to disentangle the influence of the covariance matrix eigendecomposition and the MPI communication in PS-CMA-ES. The first setup considers the standard parallel CMA-ES without swarm communication, i.e., $I_c = \infty$. The second benchmark evaluates the performance of the standard PS-CMA-ES with $I_c = 200$. Since $I_c$ is in units of generations, and increasing $S$ ($N_{\mathrm{proc}}$) also increases the number of function evaluations per generation, the number of MPI communications performed in total during the fixed 500 000 function evaluations decreases. Therefore, the third setup considers PS-CMA-ES with a constant number of MPI communication steps, independent of the swarm size $S$. This is achieved by setting $I_c = 200/S$. All three benchmarks are conducted in $n = 10, 30, 50, 100$ dimensions.

The Fortran library is compiled with the Intel Fortran compiler version 10.1 and optimization level O3, and linked against OpenMPI version 1.2.8. The tests are performed on a Gentoo 2.6.25 Linux cluster consisting of 12 compute nodes. Each node contains 2 Intel Xeon 2.8 GHz quad-core processors (8 cores per node) with 2 GB of RAM per core. The nodes are connected by a dedicated Gigabit Ethernet network, entirely reserved for MPI communication (there is a second, identical network for system communication). TORQUE and Maui are used as resource manager and queuing system, respectively. In order to assess the influence of intra- vs. inter-node MPI communication, the scheduler is instructed to assign 8 MPI processes per node. Each benchmark is repeated $r = 1, \ldots, R$ times. For each repetition $r$, we measure the elapsed wall-clock time $t_{i,r}$ on each processor core $i = 1, \ldots, N_{\mathrm{proc}}$. The overall run time $t(N_{\mathrm{proc}})$ of the algorithm on $N_{\mathrm{proc}}$ processors is given by the maximum time over all processes, averaged over the $R$ independent runs:

$$t(N_{\mathrm{proc}}) = \mathrm{mean}_r \max_{i=1,\ldots,N_{\mathrm{proc}}} t_{i,r} \,. \tag{1}$$

From this, the parallel speedup $s$ and efficiency $e$ are defined as:

$$s(N_{\mathrm{proc}}) = \frac{t(1)}{t(N_{\mathrm{proc}})} \,, \quad e(N_{\mathrm{proc}}) = \frac{s(N_{\mathrm{proc}})}{N_{\mathrm{proc}}} \,. \tag{2}$$

The measured maximum wall-clock times for all 3 benchmarks are reported in Fig. 7, the speedups in Fig. 8, and the parallel efficiencies in Fig. 9.

In $n = 10$ dimensions, there are no noticeable differences between the three different test setups. Up to $N_{\text{proc}} = 8$, i.e. on a single node, the wall-clock time decreases from 2.5s to below 0.5s. The speedup increases up to 6 and the efficiency decreases to 0.6–0.7. This should be compared to 37s for 50 000 function evaluations on $N_{\text{proc}} = 4$ using the MATLAB implementation (Hansen, 2008). The Fortran library thus is about 460 times faster than the MATLAB implementation. When using two nodes (16 processes), the wall-clock time increases again, and speedup and efficiency drop considerably. This is expected as the network latency becomes the limiting factor for such a small test problem. The situation changes in higher dimensions. For $n = 30$, the wall-clock time of parallel CMA-ES decreases from 12s on a single core to below 1s on 64 cores. The two PS-CMA-ES tests need around 17s on a single core due to the additional construction of the rotation matrix. The PS-CMA-ES with constant number of MPI communications shows a similar scaling as the parallel CMA-ES, with an offset of about 4–5 seconds, corresponding to the constant communication overhead. The PS-CMA-ES with decreasing number of communications approaches the behavior of the standard parallel CMA-ES since, with increasing $N_{\text{proc}}$, the MPI communication overhead and the 30-dimensional rotations become negligible compared to the computational cost of CMA-ES. This is also reflected in the parallel speedup and efficiency. The standard PS-CMA-ES with $I_c = 200$ achieves the best efficiency (due to a higher computational cost on a single core), closely followed by the parallel CMA-ES. The existing MATLAB implementation needed 75s for 50 000 function evaluations on $N_{\text{proc}} = 4$, thus about 200 times longer. The same qualitative behavior is observed in $n = 50$ (figures not shown), but, due to the higher computational cost, the parallel efficiency increases further. The computational costs for the basic CMA-ES operations and the matrix rotations now dominate, and the communication overhead becomes less apparent. On a single core, parallel CMA-ES needs 40s and the two PS-CMA-ES variants around 68s. While the wall-clock time of the standard PS-CMA-ES rapidly approaches the one of CMA-ES for increasing $N_{\text{proc}}$, the PS-CMA-ES with a constant number of MPI communications shows an offset of around 25s due to the communication overhead and the 50-dimensional matrix rotation. The speedups of the parallel CMA-ES and the standard PS-CMA-ES for $N_{\text{proc}} = 64$ are 40 and 50, respectively, corresponding to parallel efficiencies of 0.55 and 0.5. For comparison, the MATLAB implementation required 187s for 50 000 function evaluations and hence was about 150 times slower than the Fortran library. For $n = 100$, the parallel scaling further improves. The efficiency for standard CMA-ES is 0.87 on 64 cores, while standard PS-CMA-ES achieves a super-linear efficiency of 1.07 (due to the decreasing number of MPI communications).

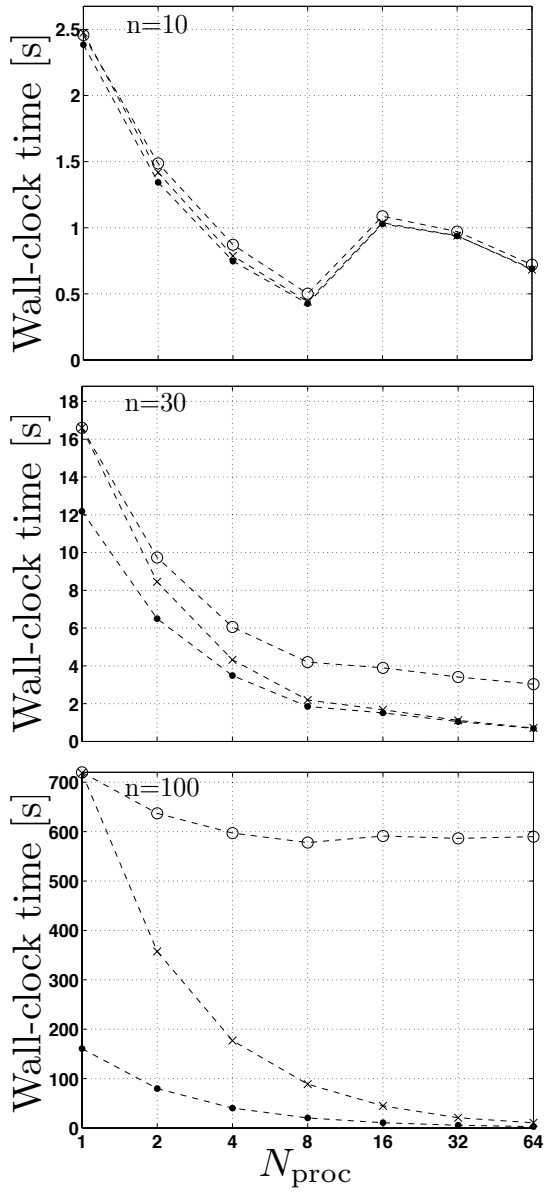



Figure 7: Overall run time $t(N_{\mathrm{proc}})$ in seconds for the parallel CMA-ES (•) and PS-CMA-ES with constant (∘) and decreasing (×) number of MPI communications on the random landscape test problem in $n = 10, 30, 100$ dimensions. The number of processor cores $N_{\mathrm{proc}}$ is varied from 1 to 64. Each point is averaged from $R = 5$ runs. The standard deviations are close to zero (data not shown).

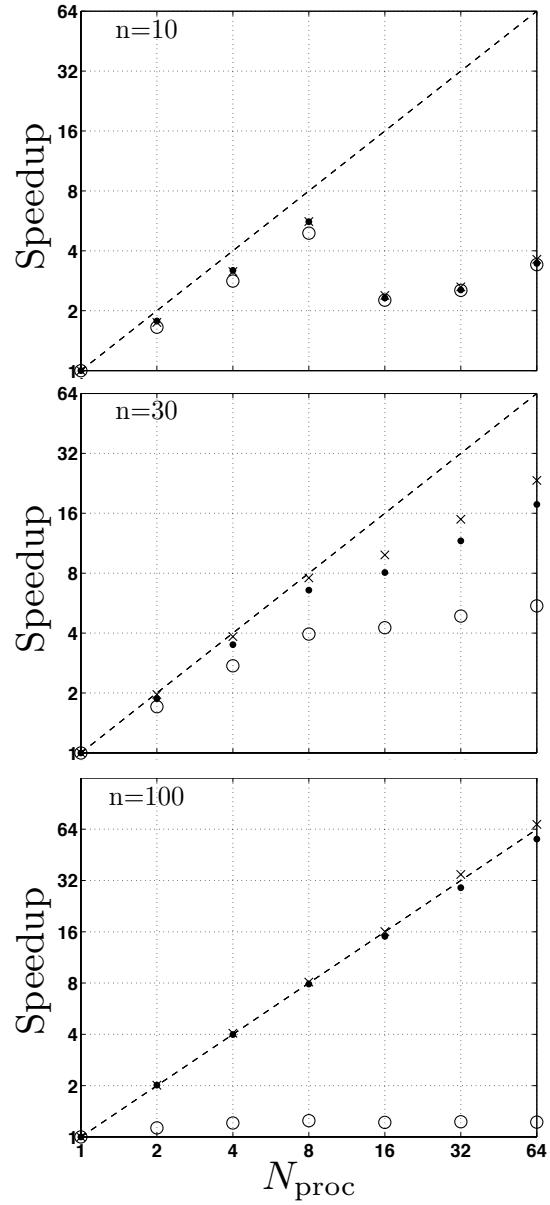



Figure 8: Parallel speedup $s$ of the parallel CMA-ES (•), and PS-CMA-ES with constant (∘) and decreasing (×) number of MPI communications on the random landscape test problem in $n = 10, 30, 100$ dimensions. The number of processor cores $N_{\mathrm{proc}}$ is varied from 1 to 64. Each point is averaged from $R = 5$ runs. The standard deviations are close to zero (data not shown).

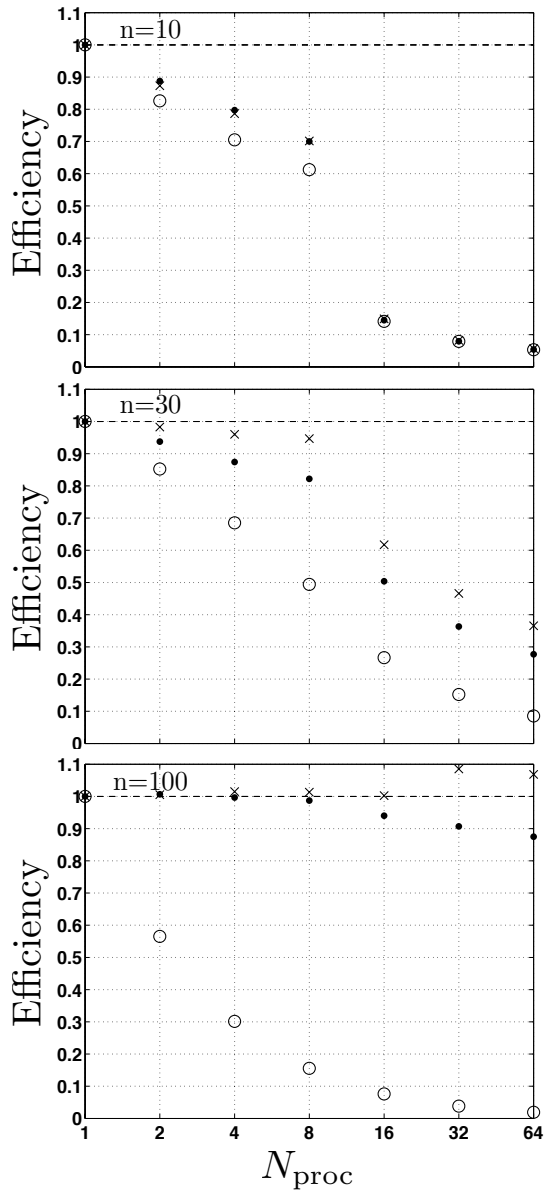



Figure 9: Parallel efficency $e$ of the parallel CMA-ES (•), and PS-CMA-ES with constant (○) and decreasing (×) number of MPI communications on the random landscape test problem in $n = 10, 30, 100$ dimensions. The number of processor cores $N_{\mathrm{proc}}$ is varied from 1 to 64. Each point is averaged from $R = 5$ runs. The standard deviations are close to zero (data not shown).

# Bibliography


Acklam, P. J. 2009. An algorithm for computing the inverse normal cumulative distribution function.

Adib, A. B. 2005. NP-hardness of the cluster minimization problem revisited. Journal of Physics A: Mathematical and General **38**:8487.

Akimoto, Y., Y. Nagata, I. Ono, and S. Kobayashi. 2011. Bidirectional Relation between CMA Evolution Strategies and Natural Evolution Strategies. In: Schaefer, R., C. Cotta, J. Kolodziej, and G. Rudolph, editors, Parallel Problem Solving from Nature – PPSN XI, volume 6238 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 154–163.

Alba, E. 2005. Parallel Metaheuristics: A New Class of Algorithms. Wiley-Interscience.

Amari, S.-i. 1998. Natural Gradient Works Efficiently in Learning. Neural Computation **10**:251–276.

Andrieu, C., and E. Moulines. 2006. On the ergodicity properties of some adaptive MCMC algorithms. Annals of Applied Probability **16**:1462–1505.

Andrieu, C., and C. P. Robert. 2001. Controlled MCMC for Optimal Sampling. Working Papers 2001-33, Centre de Recherche en Economie et Statistique.

Andrieu, C., and J. Thoms. 2008. A tutorial on adaptive MCMC. Statistics and Computing **18**:343–373.

Anfinsen, C. B. 1973. Principles that govern the folding of protein chains. Science (New York, N.Y.) **181**:223–230.

Angel, E., and V. Zissimopoulos. 2000. On the classification of NP-complete problems in terms of their correlation coefficient. Discrete Applied Mathematics **99**:261 – 277.

Arkus, N., V. N. Manoharan, and M. P. Brenner. 2009. Minimal Energy Clusters of Hard Spheres with Short Range Attractions. Phys. Rev. Lett. **103**.

Auger, A., and N. Hansen. 2005a. Performance evaluation of an advanced local search evolutionary algorithm. In: Proc. of IEEE Congress on Evolutionary Computation (CEC 2005), volume 2. 1777–1784 Vol. 2.

———. 2005b. A restart CMA evolution strategy with increasing population size. In: Proc. of IEEE Congress on Evolutionary Computation (CEC 2005), volume 2. 1769–1776.

Banavar, J. R., M. Cieplak, T. X. Hoang, and A. Maritan. 2009. First-principles design of nanomachines. Proceedings of the National Academy of Sciences **106**:6900–6903.

Banks, A., J. Vincent, and C. Anyakoha. 2007. A review of particle swarm optimization. Part I: background and development. Natural Computing **6**:467–484.

Barron, C., S. Gómez, and D. Romero. 1996. Archimedean polyhedron structure yields a lower energy atomic cluster. Applied Mathematics Letters **9**:75 – 78.

Baumgartner, B. 2008. Particle Swarm CMA-ES. Diploma thesis, Institute of Computational Science, Department of Computer Science, ETH Zürich.

Bersini, H., M. Dorigo, S. Langerman, G. Seront, and L. Gambardella. 1996. Results of the first international contest on evolutionary optimisation (1st ICEO). In: Evolutionary Computation, 1996., Proceedings of IEEE International Conference on. 611 –615.

Betancourt, M. R., and J. Skolnick. 2001. Universal similarity measure for comparing protein structures. Biopolymers **59**:305–309.

Beyer, H.-G. 2001. The theory of evolution strategies. Natural Computing. New York, NY, USA: Springer-Verlag New York, Inc.

Beyer, H.-G., and B. Sendhoff. 2008. Covariance Matrix Adaptation Revisited –The CMSA Evolution Strategy. In: Lecture Notes in Computer Science, Parallel Problem Solving from Nature –PPSN X. Springer, 123–132.

Biester, C., P. J. Grabner, G. Larcher, and R. Tichy. 1995. Adaptive search in Quasi Monte-Carlo optimization. Math. Comput. **64**:807–818.

Boese, K. D. 1995. Cost Versus Distance In the Traveling Salesman Problem. Technical report, UCLA Computer Science Dept., Losa Angeles, CA 90024-1596, USA.

Boese, K. D., A. B. Kahng, and S. Muddu. 1994. A new adaptive multi-start technique for combinatorial global optimizations. Operations Research Letters **16**:101 – 113.

Born, M., and R. Oppenheimer. 1927. Zur Quantentheorie der Molekeln. Annalen der Physik **389**:457–484.

Box, G. E. P. 1957. Evolutionary Operation: A Method for Increasing Industrial Productivity. Applied Statistics **6**:81–101.

Box, G. E. P., and M. E. Muller. 1958. A Note on the Generation of Random Normal Deviates. The Annals of Mathematical Statistics **29**:pp. 610–611.

Box, G. E. P., and K. B. Wilson. 1951. On the experimental attainment of optimum conditions. Journal of the Royal Society Series B **13**:1–45.

Boyd, S., and L. Vandenberghe. 2004. Convex Optimization. Cambridge University Press.

Braak, C. J. 2006. A Markov Chain Monte Carlo version of the genetic algorithm Differential Evolution: easy Bayesian computing for real parameter spaces. Statistics and Computing **16**:239–249.

Brooks, S. H. 1958. A Discussion of Random Methods for Seeking Maxima. Operations Research **6**:244–251.

Broyden, C. G. 1970. The Convergence of a Class of Double-rank Minimization Algorithms 1. General Considerations. IMA Journal of Applied Mathematics **6**:76–90.

Bryngelson, J., and P. Wolynes. 1989. Intermediates and Barrier Crossing in a Random Energy Model (with applications to protein folding). Journal of Physical Chemistry **93**:6902–6915.

Burkardt, J. 2009. FORTRAN90 software collection.

Caflisch, R. E., W. J. Morokoff, and A. B. Owen. 1997. Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension. J. Comp. Finance **1**:27–46.

Call, S. T., D. Y. Zubarev, and A. I. Boldyrev. 2007. Global minimum structure searches via particle swarm optimization. J. Comp. Chem. **28**:1177–1186.

Campolongo, F., J. Cariboni, A. Saltelli, and W. Schoutens. 2004. Enhancing the Morris Method. In: Hanson, K., and F. Hemez, editors, Proceedings of the 4th International Conference on Sensitivity Analysis of Model Output (SAMO 2004). Los Alamos National Laboratory, Los Alamos, U.S.A., 369–379.

Cantor, C. R., and R. Schimmel, Paul. 1980. Biophysical chemistry Part III: The behavior of biological macromolecules. W. H. Freeman, New York.

Carneiro, M., and D. L. Hartl. 2010. Adaptive landscapes and protein evolution. Proceedings of the National Academy of Sciences of the United States of America **107**:1747–1751.

Catlow, C. R. A., S. T. Bromley, S. Hamad, M. Mora-Fonz, A. A. Sokol, and S. M. Woodley. 2010. Modelling nano-clusters and nucleation. Physical Chemistry Chemical Physics **12**:786–811.

Chakrabarti, R., and H. Rabitz. 2007. Quantum control landscapes. International Reviews In Physical Chemistry **26**:671–735.

Clark, P. L. 2004. Protein folding in the cell: reshaping the folding funnel. Trends Biochem. Sci. **29**:527–534.

Cleri, F., and V. Rosato. 1993. Tight-binding potentials for transition metals and alloys. Phys. Rev. B **48**:22–33.

Cochran, A. G., N. J. Skelton, and M. A. Starovasnik. 2001. Tryptophan zippers: Stable, monomeric $\beta$-hairpins. Proceedings of the National Academy of Sciences of the United States of America **98**:5578–5583.

Cohn, H., and A. Kumar. 2009. Algorithmic design of self-assembling structures. Proceedings of the National Academy of Sciences of the United States of America **106**:9570–9575.

Cox, G., R. S. Berry, and R. L. Johnston. 2006. Characterizing potential surface topographies through the distribution of saddles and minima. Journal of Physical Chemistry a **110**:11543–11550.

Crick, F. 1970. Central Dogma of Molecular Biology. Nature **227**:561–&.

Croes, G. 1958. A Method for Solving Traveling Salesman Problems. Operations Research **6**:791–812.

Davidon, W. C. 1991. Variable Metric Method for Minimization. SIAM Journal on Optimization **1**:1–17.

Dawkins, R. 1967. The Selfish Gene. Oxford University Press.

de Araujo, A., A. Gomes, A. Bursztyn, and E. Shakhnovich. 2008. Native atomic burials, supplemented by physically motivated hydrogen bond constraints, contain sufficient information to determine the tertiary structure of small globular proteins. Proteins - Structure, Function and Bioinformatics **70**:971–983.

De Jong, K. A. 1975. An analysis of the behavior of a class of genetic adaptive systems. Ph.D. thesis, Ann Arbor, MI, USA.

Deaven, D., N. Tit, J. Morris, and K. Ho. 1996. Structural optimization of Lennard-Jones clusters by a genetic algorithm. Chem. Phys. Lett. **256**:195–200.

Dill, K. A., and H. S. Chan. 1997. From Levinthal to pathways to funnels. Nat. Struct. Mol. Biol. **4**:10–19.

Doye, J. 2000. Effect of compression on the global optimization of atomic clusters. Phys. Rev. E **62**:8753–8761.

Doye, J. P. K., M. A. Miller, and D. J. Wales. 1999a. Evolution of the potential energy surface with size for Lennard-Jones clusters. The Journal of Chemical Physics **111**:8417–8428.

———. 1999b. The double-funnel energy landscape of the 38-atom Lennard-Jones cluster. J. Chem. Phys. **110**:6896–6906.

Drew, S. S., and T. H. de Mello. 2006. Quasi-Monte Carlo strategies for stochastic optimization. In: WSC '06: Proceedings of the 38th conference on Winter simulation. Winter Simulation Conference, 774–782.

Dueck, G., and T. Scheuer. 1990. Threshold accepting: a general purpose optimization algorithm appearing superior to simulated annealing. J. Comput. Phys. **90**:161–175.

Faraldo-Gomez, J. D., and B. Roux. 2007. On the importance of a funneled energy landscape for the assembly and regulation of multidomain Src tyrosine kinases. Proceedings of the National Academy of Sciences of the United States of America **104**:13643–13648.

Faure, H. 1992. Good permutations for extreme discrepancy. Journal of Number Theory **42**:47 – 56.

Finck, S., N. Hansen, R. Ros, and A. Auger. 2009. Real-Parameter Black-Box Optimization Benchmarking 2009: Presentation of the Noiseless Functions Contents.

Finnis, M. W., and J. E. Sinclair. 1984. A simple empirical N-body potential for transition metals. Philosophical Magazine A **50**:45–55.

Fisher, H., and G. Thompson. 1963. Probabilistic learning combinations of local job-shop scheduling rules. In: Muth, J., and G. Thompson, editors, Industrial Scheduling. Englewood Cliffs, NJ: Prentice-Hall, 225–251.

Fletcher, R. 1970. A new approach to variable metric algorithms. The Computer Journal **13**:317–322.

Flory, P. J. 1953. Principles of Polymer Chemistry. Cornell University Press.

———. 1969. Statistical Mechanics of Chain Molecules. Wiley-Interscience, New York.

Frauenfelder, H. and Sligar, S. G. and Wolynes, P. G. 1991. The energy landscapes and motions of proteins. Science **254**:1598–1603.

Gallagher, M., and B. Yuan. 2006. A general-purpose tunable landscape generator. IEEE Trans. Evolutionary Computation **10**:590–603.

Gelman, A., G. Roberts, and W. Gilks. 1996. Efficient Metropolis jumping rules. In: Bernado, J. M., et al., editors, Bayesian Statistics, volume 5. OUP, 599.

Geman, S., and D. Geman. 1984. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Transactions On Pattern Analysis and Machine Intelligence **6**:721–741.

Geyer, C. J., and E. A. Thompson. 1995. Annealing Markov Chain Monte Carlo with Applications to Ancestral Inference. Journal of the American Statistical Association **90**:909–920.

Gillespie, J. H. 1984. Molecular Evolution Over the Mutational Landscape. Evolution **38**:1116–1129.

Glasmachers, T., T. Schaul, S. Yi, D. Wierstra, and J. Schmidhuber. 2010. Exponential natural evolution strategies. In: GECCO '10: Proceedings of the 12th annual conference on Genetic and evolutionary computation. New York, NY, USA: ACM, 393–400.

Glover, F. 1989. Tabu Search–Part I. INFORMS JOURNAL ON COMPUTING **1**:190–206.

Goffin, J.-L. 1984. Variable metric relaxation methods, part II: The ellipsoid method. Mathematical Programming **30**:147–162. 10.1007/BF02591882.

Goldberg, A. D., C. D. Allis, and E. Bernstein. 2007. Epigenetics: A landscape takes shape. Cell **128**:635–638.

Goldfarb, D. 1970. A Family of Variable-Metric Methods Derived by Variational Means. Mathematics of Computation **24**:pp. 23–26.

Goldstein, M. 1969. Viscous Liquids and the Glass Transition: A Potential Energy Barrier Picture. J. Chem. Phys **51**:3728–&.

Golub, G. H., and C. F. Van Loan. 1996. Matrix Computations. Johns Hopkins University Press, 3rd edition.

Graeb, H. 2009. Optimization Methods for Circuit Design. Arcisstr. 21.

Green, P., and X. Han. 1992. Metropolis methods, Gaussian proposals and antithetic variables. In: Barone, P., A. Frigessi, and M. Piccioni, editors, Stochastic Models, Statistical Methods and Algorithms in Image Analysis. Berlin, Germany: Springer-Verlag, 142–64.

Grötschel, M., L. Lovász, and A. Schrijver. 1993. Geometric Algorithms and Combinatorial Optimization, volume 2 of *Algorithms and Combinatorics*. Springer, second corrected edition edition.

Haario, H., E. Saksman, and J. Tamminen. 1999. Adaptive proposal distribution for random walk Metropolis algorithm. Computational Statistics **14**:375–395.

———. 2001. An adaptive Metropolis algorithm. Bernoulli **7**:223–242.

———. 2005. Componentwise adaptation for high dimensional MCMC. Computational Statistics **20**:265–273.

Hales, T. 2005. A proof of the Kepler conjecture. Annals of Mathematics **162**:1065–1185.

Halton, J. H. 1960. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. Numerische Mathematik **2**:84–90. 10.1007/BF01386213.

Hansen, N. 2000. Invariance, Self-Adaptation and Correlated Mutations in Evolution Strategies. In: PPSN VI: Proceedings of the 6th International Conference on Parallel Problem Solving from Nature. London, UK: Springer-Verlag, 355–364.

———. 2006. Compilation of Results on the 2005 CEC Benchmark Function Set. Technical report, Computational Laboratory (CoLab), Institute of Computational Science, ETH Zurich.

———. 2008. The CMA Evolution Strategy: A Tutorial. http://www.lri.fr/ hansen/cmatutorial.pdf.

———. 2009a. Benchmarking the Nelder-Mead downhill simplex algorithm with many local restarts. In: GECCO '09: Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference. New York, NY, USA: ACM, 2403–2408.

———. 2009b. http://www.lri.fr/ hansen/cmaapplications.pdf.

———. 2010a. Personal communication.

———. 2010b. Variable Metrics in Evolutionary Computation. Habilitation thesis, Université Paris-Sud 11.

Hansen, N., and S. Kern. 2004. Evaluating the CMA Evolution Strategy on Multimodal Test Functions. In: Lecture Notes in Computer Science, Parallel Problem Solving from Nature – PPSN VIII. Springer, 282–291.

Hansen, N., S. D. Muller, and P. Koumoutsakos. 2003. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evol. Comput. **11**:1–18.

Hansen, N., A. Niederberger, L. Guzzella, and P. Koumoutsakos. 2009. A Method for Handling Uncertainty in Evolutionary Optimization With an Application to Feedback Control of Combustion. Evolutionary Computation, IEEE Transactions on **13**:180 –197.

Hansen, N., and A. Ostermeier. 1996. Adapting Arbitrary Normal Mutation Distributions in Evolution Strategies: The Covariance Matrix Adaptation. In: Proceedings of the 1996 IEEE Conference on Evolutionary Computation (ICEC '96). 312–317.

———. 2001. Completely Derandomized Self-Adaption in Evolution Strategies. Evolutionary Computation **9**:159–195.

Hansen, N., A. Ostermeier, and A. Gawelczyk. 1995. On the Adaptation of Arbitrary Normal Mutation Distributions in Evolution Strategies: The Generating Set Adaptation. In: Proceedings of the 6th International Conference on Genetic Algorithms. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 57–64.

Hart, W. E., and S. Istrail. 1997. Robust Proofs of NP-Hardness for Protein Folding: General Lattices and Energy Potentials. Journal of Computational Biology **4**:1–22.

Hartke, B. 2004. Application of evolutionary algorithms to global cluster geometry optimization. Applications of Evolutionary Computation In Chemistry **110**:33–53.

Hasse, R. W. 1991. Structure and magic numbers of large Lennard-Jones quasicrystals and crystals. Physics Letters A **161**:130 – 134.

Hastings, W. 1970. Monte Carlo sampling methods using Markov chains and their applications. Biometrika **57**:97–109.

He, J., C. Reeves, C. Witt, and X. Yao. 2007. A note on problem difficulty measures in black-box optimization: Classification, realizations and predictability. Evol. Comput. **15**:435–443.

Hoare, M. R. 1979. Structure and Dynamics of Simple Microclusters. John Wiley & Sons, Inc.

Hooke, R., and T. A. Jeeves. 1961. "Direct Search" Solution of Numerical and Statistical Problems. J. ACM **8**:212–229.

Horn, B. 1987. Closed-form solution of absolute orientation using unit quaternions. Journal of the Optical Society of America A **4**:629.

Horn, B., M. Hilden, and S. Negahdaripour. 1988. Closed-form solution of absolute orientation using orthonormal matrices. Journal of the Optical Society of America A **5**:1127.

Hsieh, C.-T., C.-M. Chen, and Y.-P. Chen. 2007. Particle Swarm Guided Evolution Strategy. In: Lipson, H., editor, Genetic and Evolutionary Computation Conference (GECCO '07). London, England, 650–657.

Hsieh, M., R. Wu, and H. Rabitz. 2009. Topology of the quantum control landscape for observables. Journal of Chemical Physics **130**.

Hsieh, M., R. Wu, C. Rosenthal, and H. Rabitz. 2008. Topological and statistical properties of quantum control transition landscapes. Journal of Physics B-Atomic Molecular and Optical Physics **41**.

Hu, B., and K.-W. Tsui. 2008. Distributed evolutionary Monte Carlo for Bayesian computing. Computational Statistics & Data Analysis **In Press, Corrected Proof**:–.

Hu, T. C., V. Klee, and D. Larman. 1989. Optimization of Globally Convex Functions. SIAM Journal on Control and Optimization **27**:1026–1047.

Hunjan, J., A. Tovchigrechko, Y. Gao, and I. A. Vakser. 2008. The size of the intermolecular energy funnel in protein-protein interactions. Proteins-Structure Function and Bioinformatics **72**:344–352.

Igel, C., T. Suttorp, and N. Hansen. 2006. A computational efficient covariance matrix update and a (1+1)-CMA for evolution strategies. In: GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation. New York, NY, USA: ACM, 453–460.

Ikeda, K., and S. Kobayashi. 2000. GA Based on the UV-Structure Hypothesis and Its Application to JSP. In: PPSN VI: Proceedings of the 6th International Conference on Parallel Problem Solving from Nature. London, UK: Springer-Verlag, 273–282.

Jaynes, E. T. 1957. Information Theory and Statistical Mechanics. Phys. Rev. **106**:620–630.

Jens Jägersküpper. 2008. Lower bounds for randomized direct search with isotropic sampling. Operations Research Letters **36**:327 – 332.

Jones, T., and S. Forrest. 1995. Fitness Distance Correlation as a Measure of Problem Difficulty for Genetic Algorithms. In: Proceedings of the 6th International Conference on Genetic Algorithms. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 184–192.

Joy, C., P. Boyle, and K. Tan. 1996. Quasi-Monte Carlo methods in numerical finance. Management Science **42**:926–938.

Kabsch, W. 1976. A solution for the best rotation to relate two sets of vectors. Acta Crystallogr. A. **32**:922–923.

———. 1978. A discussion of the solution for the best rotation to relate two sets of vectors. Acta Crystallogr. A. **34**:827–828.

Kaindl, K., and B. Steipe. 1997. Metric properties of the root-mean-square deviation of vector sets. Acta Crystallographica A **53**:809.

Kaplan, J. 2008. The end of the adaptive landscape metaphor? Biology and Philosophy **23**:625–638.

Kapon, R., R. Nevo, and Z. Reich. 2008. Protein energy landscape roughness. Biochemical Society Transactions **36**:1404–1408.

Kauffman, S. A. 1993. The Origins of Order: Self-Organization and Selection in Evolution. Oxford University Press, USA, 1 edition.

Kauffman, S. A., and E. D. Weinberger. 1989. The NK model of rugged fitness landscapes and its application to maturation of the immune response. Journal of Theoretical Biology **141**:211 – 245.

Kauffman, W. E., Stuart A., and A. S. Perelson. 1988. Maturation of the Immune Response Via Adaptive Walks On Affinity Landscapes. In: Perelson, A. S., editor, Theoretical Immunology: Part One, volume 1. New York: Addison-Wesley, 349—382.

Kennedy, J., and R. Eberhart. 1995. Particle swarm optimization. Proc. IEEE, International Conference on Neural Networks **4**:1942–1948.

Khachiyan, L. 1979. A polynomial Algorithm in Linear Programming. Doklady Akademiia Nauk SSSR **244**:1093–1096.

Kim, J., and T. Keyes. 2007. Inherent structure analysis of protein folding. Journal of Physical Chemistry B **111**:2647–2657.

Kimura, M. 1983. The Neutral Theory of Molecular Evolution. New York: Cambridge University Press.

Kirkpatrick, S., C. Gelatt, and M. P. Vecchi. 1983. Optimization by Simulated Annealing. Science **220**:671–680.

Kjellström, G. 1969. Network Optimization by Random Variation of Component Values. Ericsson Technics **25**:133–151.

———. 1970. Optimization of Electrical Networks with Respect to Tolerance Costs. Ericsson Technics **26**:157–175.

———. 1991. On the Efficiency of Gaussian Adaptation. J. Optim. Theory Appl. **71**.

Kjellström, G., and L. Taxen. 1981. Stochastic Optimization in System Design. IEEE Trans. Circ. and Syst. **28**.

Kjellström, G., and L. Taxen. 1992. Gaussian Adaptation, an evolution-based efficient global optimizer. In: Comp. Appl. Math. Elsevier Science, 267—276.

Kneller, G. R. 1991. Superposition of Molecular Structures using Quaternions. Mol. Simulat. **7**:113–119.

———. 2005. Comment on "Using quaternions to calculate RMSD" - [J. Comp. Chem. 25, 1849 (2004)]. J. Comp. Chem. **26**:1660–1662.

König, M. 2010. Design and Implementation of Parallel Island Models for CMA-ES. Master's thesis, Institute of Theoretical Computer Science.

Kramer, O. 2010. A Review of Constraint-Handling Techniques for Evolution Strategies. Applied Computational Intelligence and Soft Computing .

Krykova, I. 2003. Evaluating of path-dependent secutirties with low discrepancy methods. Master's thesis, Worcester Polytechnic Institute.

Kucherenko, S., and Y. Sytsko. 2005. Application of deterministic low-discrepancy sequences in global optimization. Comp. Optim. Appl. **30**:297–318.

Leary, R. 2000. Global optimization on funneling landscapes. Journal of Global Optimization **18**:367–383.

Leary, R. H. 1997. Global Optima of Lennard-Jones Clusters. Journal of Global Optimization **11**:35–53. 10.1023/A:1008276425464.

Lennard-Jones, J. 1924. On the determination of molecular fields - II From the equation of state of a gas. Proceedings of the Royal Society of London Series A-Containing Papers of a Mathematical and Physical Character **106**:463–477.

Li, Z., and H. A. Scheraga. 1987. Monte Carlo-minimization approach to the multiple-minima problem in protein folding. Proceedings of the National Academy of Sciences of the United States of America **84**:6611–6615.

Liang, F., and W. Wong. 2001a. Evolutionary Monte Carlo for protein folding simulations. Journal of Chemical Physics **115**:3374–3380.

———. 2001b. Real-Parameter Evolutionary Monte Carlo With Applications to Bayesian Mixture Models. Journal of the American Statistical Association **96**:653–666.

Lin, S., and B. W. Kernighan. 1973. An Effective Heuristic Algorithm for the Traveling-Salesman Problem. Operations Research **21**:498–516.

Liu, J. S. 2002. Monte Carlo Strategies in Scientific Computing. Springer.

Liu, R., and A. Owen. 2006. Estimating mean dimensionality of analysis of variance decompositions. J. Am. Stat. Assoc. **101**:712–721.

Lovász, L. 1999. Hit-and-run mixes fast. Mathematical Programming **86**:443–461. 10.1007/s101070050099.

Lunacek, M., and D. Whitley. 2006. The Dispersion Metric and the CMA Evolution Strategy. In: GECCO '06: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation. New York, USA: ACM Press, 477–484.

Lunacek, M., D. Whitley, and A. Sutton. 2008. The Impact of Global Structure on Search. In: Lect. Notes Comput. Sc., volume 5199 of *Parallel Problem Solving from Nature - PPSN X*. 498–507.

Maiorov, V. N., and G. M. Crippen. 1994. Significance of Root-Mean-Square Deviation in Comparing 3-Dimenisonal Structures of Globular Proteins. J. Mol. Biol. **235**:625–634.

———. 1995. Size-Independent comparison of Protein 3-Dimensional Structures. Proteins: Struct. Funct. Genet. **22**:273–283.

Marinari, E., and G. Parisi. 1992. Simulated Tempering: A New Monte Carlo Scheme. EPL (Europhysics Letters) **19**:451.

Matsumoto, M., and T. Nishimura. 1998. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Trans. Model. Comput. Simul. **8**:3–30.

Maynard-Smith, J. 1970. Natural Selection and the Concept of a Protein Space. Nature **225**:563–564.

McLachlan, A. D. 1972. Mathematical procedure for superimpsoing atomic coordinates of proteins. Acta Crystallogr. A. **A 28**:656–657.

———. 1979. Gene duplications in the structural evolution of Chymotrypsin. J. Mol. Biol. **128**:49–74.

———. 1984. How alike are the Shapes of Two Random Chains. Biopolymers **23**:1325–1331.

McLeish, D. L. 1975. A Maximal Inequality and Dependent Strong Laws. The Annals of Probability **3**:829–839.

Mello, C., and D. Barrick. 2004. An experimentally determined protein folding energy landscape. Proceedings of the National Academy of Sciences of the United States of America **101**:14102–14107.

Meng, G., N. Arkus, M. P. Brenner, and V. N. Manoharan. 2010. The Free-Energy Landscape of Clusters of Attractive Hard Spheres. Science **327**:560–563.

Merz, P. 2004. Advanced fitness landscape analysis and the performance of memetic algorithms. Evol. Comput. **12**:303–325.

———. 2005. NK-Fitness Landscapes and Memetic Algorithms with Greedy Operators and k-opt Local Search. Recent Advances in Memetic Algorithms :209–228.

Merz, P., and B. Freisleben. 1998. Memetic Algorithms and the Fitness Landscape of the Graph Bi-Partitioning Problem. Parallel Problem Solving from Nature —PPSN V :765–.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. 1953. Equation of State Calculations by Fast Computing Machines. The Journal of Chemical Physics **21**:1087–1092.

Mikki, S. M., and A. A. Kishk. 2007. Physical theory for particle swarm optimization. Prog. Electromagn. Res. **75**:171–207.

Misteli, Y., and G. Ofenbeck. 2010. BFGS in memetic CMA-ES. Technical report, Institute of Theoretical Computer Science, ETH Zürich, Zürich.

Moore, K., M. Hsieh, and H. Rabitz. 2008. On the relationship between quantum control landscape structure and optimization complexity. The Journal of Chemical Physics **128**:154117.

Moro, B. 1995. The Full Monte. Risk **8**.

Morokoff, W. J., and R. Caflisch. 1995. Quasi-Monte Carlo integration. J. Comput. Phys. **122**:218–230.

Morris, M. D. 1991. Factorial sampling plans for preliminary computational experiments. Technometrics **33**:161–174.

Morse, P. 1929. Diatomic molecules according to the wave mechanics. II. Vibrational levels. Physical Review **34**:57–64.

Müller, C. L. 2010. Exploring the common concepts of adaptive MCMC and Covariance Matrix Adaptation schemes. In: Auger, A., J. Shapiro, L. D. Whitley, and C. Witt, editors, 10361 Abstracts Collection – Theory of Evolutionary Algorithms, number 10361 in Dagstuhl Seminar Proceedings. Dagstuhl, Germany: Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

Müller, C. L., B. Baumgartner, G. Ofenbeck, B. Schrader, and I. F. Sbalzarini. 2009a. pCMALib: a parallel fortran 90 library for the evolution strategy with covariance matrix adaptation. In: GECCO '09: Proceedings of the 11th Annual conference on Genetic and evolutionary computation. New York, NY, USA: ACM, 1411–1418.

Müller, C. L., B. Baumgartner, and I. F. Sbalzarini. 2009b. Particle Swarm CMA Evolution Strategy for the optimization of multi-funnel landscapes. In: Proc. of IEEE Congress on Evolutionary Computation (CEC 2009). 2685–2692.

Müller, C. L., G. Paul, and I. F. Sbalzarini. 2007. Sensitivities for Free: CMA-ES based Sensitivity Analysis. In: Abstracts of the 5th International Conference on Sensitivity Analysis of Model Output. Budapest, Hungary, 123–124.

Müller, C. L., and I. F. Sbalzarini. 2009. A Tunable Real-world Multi-funnel Benchmark Problem for Evolutionary Optimization - And Why Parallel Island Models Might Remedy the Failure of CMA-ES on It. In: Dourado, A., A. C. Rosa, and K. Madani, editors, Proc. of the International Joint Conference on Computational Intelligence (IJCCI). INSTICC Press, 248–253.

———. 2010a. Energy landscapes of atomic clusters as black-box optimization benchmarks. submitted to Evolutionary Computation .

———. 2010b. Gaussian Adaptation as a unifying framework for continuous black-box optimization and adaptive Monte Carlo sampling. In: Evolutionary Computation (CEC), 2010 IEEE Congress on. 1 –8.

———. 2010c. Gaussian adaptation revisited - an entropic view on covariance matrix adaptation. In: C. Di Chio et al., editor, EvoApplications, volume I of *Lecture Notes in Computer Science*. Springer, 432–441.

Müller, C. L., I. F. Sbalzarini, W. F. van Gunsteren, B. Zagrovic, and P. H. Huenenberger. 2009. In the eye of the beholder: Inhomogeneous distribution of high-resolution shapes within the random-walk ensemble. J. Chem. Phys. **130**.

Müller, K., and L. D. Brown. 1979. Location of saddle points and minimum energy paths by a constrained simplex optimization procedure. Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta) **53**:75–93. 10.1007/BF00547608.

Müller, S. D., N. N. Schraudolph, and P. Koumoutsakos. 2003. Evolutionary and Gradient-Based Algorithms for Lennard-Jones Cluster Optimization. In: GECCO '03: Proceedings of the 5th Annual Conference on Genetic and Evolutionary Computation.

Neal, R. 2003. Slice sampling. Annals of Statistics **31**:705–741.

Nelder, J. A., and R. Mead. 1965. A Simplex Method for Function Minimization. The Computer Journal **7**:308–313.

Nevo, R., V. Brumfeld, R. Kapon, P. Hinterdorfer, and Z. Reich. 2005. Direct measurement of protein energy landscape roughness. Embo Reports **6**:482–486.

Nishikawa, T., and A. E. Motter. 2010. Network synchronization landscape reveals compensatory structures, quantization, and the positive effect of negative interactions. Proceedings of the National Academy of Sciences of the United States of America **107**:10342–10347.

Nocedal, J. 1980. Updating Quasi-Newton Matrices with Limited Storage. Mathematics of Computation **35**:pp. 773–782.

Northby, J. 1987. Structure and binding of Lennard-Jones clusters: $13 \leq N \leq 147$. Journal of Chemical Physics **87**:6166–6177.

Ofenbeck, G. 2009. CMA-ES and its application to atomic cluter landscapes. Master's thesis, Institute of Theoretical Computer Science, ETH Zürich, Zürich.

Oldfield, T. J., and R. E. Hubbard. 1994. Analysis of C-alpha geometry in protein structures. Proteins: Struct. Funct. Genet. **18**:324–337.

Osada, R., T. Funkhouser, B. Chazelle, and D. Dobkin. 2002. Shape Distributions. ACM Trans. Graph. **21**:807–832.

Padberg, M., and G. Rinaldi. 1987. Optimization of a 532-city symmetric traveling salesman problem by branch and cut. Operations Research Letters **6**:1 – 7.

Papadimitriou, C. H., and K. Steiglitz. 1998. Combinatorial Optimization : Algorithms and Complexity. Dover Publications.

Pines, D., editor. 1988. Emerging syntheses in science : proceedings of the founding workshops of the Santa Fe Institute, 1. Addison-Wesley.

Powell, M. 2002. UOBYQA: unconstrained optimization by quadratic approximation. Mathematical Programming **92**:555–582.

Powell, M. D. 2006. The NEWUOA software for unconstrained optimization without derivatives. In: Pardalos, P., G. Pillo, and M. Roma, editors, Large-Scale Nonlinear Optimization, volume 83 of *Nonconvex Optimization and Its Applications*. Springer US, 255–297.

Rammal, R., G. Toulouse, and M. Virasoro. 1986. Ultrametricity for physicists. Reviews of Modern Physics **58**:765–788.

Rao, F., and M. Karplus. 2010. Protein dynamics investigated by inherent structure analysis. Proceedings of the National Academy of Sciences of the United States of America **107**:9152–9157.

Rastrigin, L. A. 1963. The convergence of the random search method in the extremal control of a many-parameter system. Automation and Remote Control :1337–1342.

———. 1972. Problems of random search. Radiophysics and Quantum Electronics **15**:747–754. 10.1007/BF01031982.

Rechenberg, I. 1973. Evolutionsstrategie; Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. Stuttgart–Bad Cannstatt: Frommann-Holzboog.

Rechtsman, M., F. Stillinger, and S. Torquato. 2006a. Designed interaction potentials via inverse methods for self-assembly. Phys. Rev. E **73**:011406.

Rechtsman, M. C., F. H. Stillinger, and S. Torquato. 2005. Optimized Interactions for Targeted Self-Assembly: Application to a Honeycomb Lattice. Phys. Rev. Lett. **95**:228301.

———. 2006b. Self-assembly of the simple cubic lattice with an isotropic potential. Physical Review E **74**.

———. 2007. Synthetic diamond and wurtzite structures self-assemble with isotropic pair interactions. Physical Review E **75**.

Reidys, C., and P. Stadler. 2002. Combinatorial landscapes. Siam Review **44**:3–54.

Ren, Y., Y. Ding, and F. Liang. 2008. Adaptive evolutionary Monte Carlo algorithm for optimization with applications to sensor placement problems. Statistics and Computing **18**:375–390.

Reva, B., A. Finkelstein, and J. Skolnick. 1998. What is the probability of a chance prediction of a protein structure with an rmsd of 6 angstrom? Folding & Design **3**:141–147.

Robbins, H., and S. Monro. 1951. A Stochastic Approximation Method. The Annals of Mathematical Statistics **22**:400–407.

Rokyta, D. R., P. Joyce, S. B. Caudle, and H. A. Wichman. 2005. An empirical test of the mutational landscape model of adaptation using a single-stranded DNA virus. Nat. Genet. **37**:441–4.

Romero, P. A., and F. H. Arnold. 2009. Exploring protein fitness landscapes by directed evolution. Nature Reviews Molecular Cell Biology **10**:866–876.

Ros, Raymond and Hansen, Nikolaus. 2008. A Simple Modification in CMA-ES Achieving Linear Time and Space Complexity. In: Rudolph, G., T. Jansen, S. Lucas, C. Poloni, and N. Beume, editors, Parallel Problem Solving from Nature – PPSN X, volume 5199 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 296–305.

Rowe, W., M. Platt, D. C. Wedge, P. J. Day, D. B. Kell, and J. Knowles. 2010. Analysis of a complete DNA-protein affinity landscape. Journal of The Royal Society Interface **7**:397–408.

Rudolph, G. 1992. On Correlated Mutations in Evolution Strategies. In: Männer, R., and B. Manderick, editors, Parallel Problem Solving from Nature 2 (Proc. 2nd Int. Conf. on Parallel Problem Solving from Nature, Brussels 1992). Amsterdam: Elsevier, 105–114.

Sakuma, J., and S. Kobayashi. 2001. Extrapolation-directed crossover for real-coded GA: overcoming deceptive phenomena by extrapolative search. In: Evolutionary Computation, 2001. Proceedings of the 2001 Congress on, volume 1. 655 –662 vol. 1.

Saltelli, A., K. Chan, and E. M. Scott. 2000. Sensitivity Analysis. J. Wiley & Sons.

Schlier, C. 2004. Error trends in Quasi-Monte Carlo integration. Comp. Phys. Commun. **159**:93–105.

Schumer, M., and K. Steiglitz. 1968. Adaptive step size random search. Automatic Control, IEEE Transactions on **13**:270 – 276.

Schuster, P., and P. Stadler. 1994. Landscapes: Complex Optimization Problems and Biopolymer Structures. Computers & Chemistry **18**:295–324.

Schwefel, H.-P. 1975. Evolutionsstrategie und numerische Optimierung. Ph.D. thesis, Technical University of Berlin.

Schwefel, H.-P. P. 1993. Evolution and Optimum Seeking: The Sixth Generation. New York, NY, USA: John Wiley & Sons, Inc.

Shanno, D. F. 1970. Conditioning of Quasi-Newton Methods for Function Minimization. Mathematics of Computation **24**:pp. 647–656.

Sherrington, D. 1997. Landscape paradigms in physics and biology: Introduction and overview. Physica D: Nonlinear Phenomena **107**:117 – 121. 16th Annual International Conference of the Center for Nonlinear Studies.

Shor, N. Z. 1970. Utilization of the Operation of Space Dilatation in Minimization of Convex Functions. Kibernetika **6**:6–12.

———. 1976. Cut-off Method with Space Extension in Convex Programming Problems. Kibernetika **13**:94–95.

Sloane, N., R. Hardin, T. Duff, and J. Conway. 1995. Minimal-energy clusters of hard spheres. Discrete and Computational Geometry **14**:237–259. 10.1007/BF02570704.

Sloane, N. J. A., R. H. Hardin, T. S. Duff, and J. H. Conway. 1997. The Sphere Packing Cluster Database.

Sloane, N. J. A., R. H. Hardin, and W. D. Smith. 2000. Spherical Codes: Nice arrangements of points on a sphere in various dimensions.

Sobol, I. M. 1967. Distribution of points in a cube and approximate evaluation of integrals. Comput. Maths. Math. Phys. **7**:86–112.

Stadler, B., and P. Stadler. 2002. Generalized topological spaces in evolutionary theory and combinatorial chemistry. Journal of Chemical Information and Computer Sciences **42**:577–585.

Stadler, P. 1995. Towards a theory of landscapes. Complex Systems and Binary Networks :78–163.

Stadler, P. F. 1996. Landscapes and their correlation functions. Journal of Mathematical Chemistry **20**:1–45.

Steinhardt, P. J., D. R. Nelson, and M. Ronchetti. 1983. Bond-orientational order in liquids and glasses. Phys. Rev. B **28**:784–805.

Steipe, B. 2002a. Erratum: Metric properties of the root-mean-square deviation of vector sets. Acta Crystallographica A **58**:507.

———. 2002b. A revised proof of the metric properties of optimally superimposed vector sets. Acta Crystallographica A **58**:506.

Stillinger, F. H., and T. A. Weber. 1984. Packing Structures and Transitions in Liquids and Solids. Science **225**:983–989.

Suganthan, P. N., N. Hansen, J. J. Liang, K. Deb, Y.-P. Chen, A. Auger, and S. Tiwari. 2005. Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization. Technical report, Nanyang Technological University, Singapore.

Sullivan, D. C., T. Aynechi, V. A. Voelz, and I. D. Kuntz. 2003. Information content of molecular structures. Biophys. J. **85**:174–190.

Sullivan, D. C., and I. D. Kuntz. 2001. Conformation spaces of proteins. Proteins: Struct. Funct. and Genet. **42**:495–511.

———. 2004. Distributions in protein conformation space: Implications for structure prediction and entropy. Biophys. J. **87**:113–120.

Sun, Y., D. Wierstra, T. Schaul, and J. Schmidhuber. 2009. Efficient natural evolution strategies. In: GECCO '09: Proceedings of the 11th Annual conference on Genetic and evolutionary computation. New York, NY, USA: ACM, 539–546.

Sutton, A. P., and J. Chen. 1990. Long-range Finnis–Sinclair potentials. Philosophical Magazine Letters **61**:139–146.

Suttorp, T., N. Hansen, and C. Igel. 2009. Efficient covariance matrix update for variable metric evolution strategies. Mach. Learn. **75**:167–197.

Teytaud, O. 2008. When Does Quasi-random Work? In: PPSN X: Proceedings of the 10th International Conference on Parallel Problem Solving from Nature. Berlin, Heidelberg: Springer-Verlag, 325–336.

Teytaud, O., and S. Gelly. 2007. DCMA: yet another derandomization in covariance-matrix-adaptation. In: GECCO '07: Proceedings of the 9th annual conference on Genetic and evolutionary computation. New York, NY, USA: ACM, 955–963.

Thomson, J. J. 1904. On the Structure of the Atom: an Investigation of the Stability and Periods of Oscillation of a number of Corpuscles arranged at equal intervals around the Circumference of a Circle; with Application of the Results to the Theory of Atomic Structure. Phil. Mag. **7**:237–265.

Torquato, S. 2009. Inverse optimization techniques for targeted self-assembly. Soft Matter **5**:1157–1173.

Van Hoyweghen, C., and B. Naudts. 2000. Symmetry in the search space. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2000), volume 2. 1072 –1078 vol.2.

Van Hoyweghen, C., B. Naudts, and D. Goldberg. 2002. Spin-flip symmetry and synchronization. Evolutionary Computation **10**:317–344.

Vanneschi, L., D. Codecasa, and G. Mauri. 2010. An empirical comparison of parallel and distributed particle swarm optimization methods. In: Proceedings of the 12th annual conference on Genetic and evolutionary computation, GECCO '10. New York, NY, USA: ACM, 15–22.

Vasile, M. 2010. Personal communication.

Vempala, S. 2005. Geometric random walks: a survey. MSRI Volume on Combinatorial and Computational Geometry **52**:577–616.

Vrugt, J., B. Robinson, and J. Hyman. 2009. Self-Adaptive Multimethod Search for Global Optimization in Real-Parameter Spaces. Evolutionary Computation, IEEE Transactions on **13**:243 –259.

Vrugt, J. A., and B. A. Robinson. 2007. Improved evolutionary optimization from genetically adaptive multimethod search. Proceedings of the National Academy of Sciences of the United States of America **104**:708–711.

Waddington, C. H. 1942. The epigenotype. Endeavour **1**:18–20.

———. 1957. The Strategy of the Genes: a Discussion of Some Aspects of Theoretical Biology. London: George Allen & Unwin.

Wales, D. 2004. Energy Landscapes : Applications to Clusters, Biomolecules and Glasses (Cambridge Molecular Science). Cambridge University Press.

Wales, D. J. 2005. Energy landscapes and properties of biomolecules. Phys. Biol. **2**:S86–S93.

Wales, D. J., and J. P. K. Doye. 1997. Global Optimization by Basin-Hopping and the Lowest Energy Structures of Lennard-Jones Clusters Containing up to 110 Atoms. J. Phys. Chem. A **101**:5111–5116.

Wales, D. J., J. P. K. Doye, A. Dullweber, M. P. Hodges, F. Y. Naumkin, F. Calvo, J. Hernández-Rojas, and T. F. Middleton. 2009. The Cambridge Cluster Database.

Wales, D. J., and H. Scheraga. 1999. Review: Chemistry - Global optimization of clusters, crystals, and biomolecules. Science **285**:1368–1372.

Wang, Y., and B. Li. 2008. Understand behavior and performance of Real Coded Optimization Algorithms via NK-linkage model. In: Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on. 801 –808.

Weber, T. A., and F. H. Stillinger. 1985. Interactions, local order, and atomic-rearrangement kinetics in amorphous nickel-phosphorous alloys. Phys. Rev. B **32**:5402–5411.

Weinberger, E. D. 1990. Correlated and Uncorrelated Fitness Landscapes and How to Tell the Difference. Biological Cybernetics **63**:325–336.

———. 1996. NP Completeness of Kauffman's N-k Model, A Tuneable Rugged Fitness Landscape. Working papers, Santa Fe Institute.

Weisstein, E. W. 2009. Hypercube Line Picking. From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/HypercubeLinePicking.html.

———. 2010. Wigner 3j-Symbol. From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/Wigner3j-Symbol.html.

Whitley, D. 2010. Personal communication.

Whitley, D., K. Mathias, S. Rana, and J. Dzubera. 1995. Building Better Test Functions. In: Proceedings of the Sixth International Conference on Genetic Algorithms. Morgan Kaufmann, 239–246.

Wierstra, D., T. Schaul, J. Peters, and J. Schmidhuber. 2008. Natural Evolution Strategies. 3381 –3387.

Wille, L. 1987. Minimum-energy configurations of atomic clusters: new results obtained by simulated annealing. Chemical Physics Letters **133**:405–410.

Wille, L. T., and J. Vennik. 1985. Computational complexity of the ground-state determination of atomic clusters. Journal of Physics A: Mathematical and General **18**:L419.

Wolpert, D., and W. Macready. 1997. No free lunch theorems for optimization. "Evolutionary Computation, IEEE Transactions on" **1**:67–82.

Wolynes, P. 2001. Landscapes, funnels, glasses, and folding: From metaphor to software. Proceedings of the american Philosophical Society **145**:555–563. Joint Meeting of the Royal-Society/British-Academy, Philadelphia, Pennsylvania, APR 25-28, 2001.

Wright, S. 1932. The roles of mutation, inbreeding, crossbreeding, and selection in evolution. Proceedings of the Sixth International Congress on Genetics .

Wu, C. W., and D. Verma. 2008. A sensor placement algorithm for redundant covering based on Riesz energy minimization. In: ISCAS. 2074–2077.

Yudin, D. B., and A. S. Nemirovski. 1976. Informational complexity and effective methods for the solution of convex extremal problems. Ekonom. Mat. Metody :550–559.

Zagrovic, B., C. Snow, M. Shirts, and V. Pande. 2002. Simulation of folding of a small $\alpha$-helical protein in atomistic detail using worldwide-distributed computing. Journal of Molecular Biology **323**:927–937.

Zhou, H. 2003. Network landscape from a Brownian particle's perspective. Physical Review E **67**.

Zhou, Q., and W. H. Wong. 2008. Reconstructing the energy landscape of a distribution from Monte Carlo samples. Annals of Applied Statistics **2**:1307–1331.

# Index

# Publications

## Journal Publications

C. L. Müller and I. F. Sbalzarini. **Energy landscapes of atomic clusters as black-box optimization benchmarks.** submitted to Evolutionary Computation, 2010.

C. L. Müller, I. F. Sbalzarini, W. F. van Gunsteren, B. Žagrović, and P. H. Hünenberger. **In the eye of the beholder: Inhomogeneous distribution of high-resolution shapes within the random-walk ensemble.** J. Chem. Phys., 130(21):214904, 2009.

## Peer-reviewed Conference Publications

C. L. Müller and I. F. Sbalzarini. **Global characterization of the CEC 2005 fitness landscapes using fitness-distance analysis.** EvoApplications, Torino, Italy, April 2011. Lecture Notes in Computer Science 2011.

C. L. Müller and I. F. Sbalzarini. **Gaussian Adaptation as a unifying framwork for black-box optimization and adaptive Monte Carlo sampling.** In Proc. IEEE Congress on Evolutionary Computation (CEC), Barcelona, Spain, July 2010.

C. L. Müller and I. F. Sbalzarini. **Gaussian adaptation revisited – an entropic view on covariance matrix adaptation.** EvoApplications, Istanbul, Turkey, April 2010. Lecture Notes in Computer Science 2010.

C. L. Müller and I. F. Sbalzarini. **A tunable real-world multi-funnel benchmark problem for evolutionary optimization (and why parallel island models might remedy the failure of CMA-ES on it).** In Proc. Intl. Conf. Evolutionary Computation (ICEC), Madeira, Portugal, October 2009.

C. L. Müller, B. Baumgartner, G. Ofenbeck, B. Schrader, and I. F. Sbalzarini. **pCMALib: a parallel FORTRAN 90 library for the evolution strategy with covariance matrix adaptation.** In Proc. ACM Genetic and Evolutionary Computation Conference (GECCO '09), Montreal, Canada, July 2009.

C. L. Müller, B. Baumgartner, and I. F. Sbalzarini. **Particle swarm CMA evolution strategy for the optimization of multi-funnel landscapes.** In Proc. IEEE Congress on

Evolutionary Computation (CEC), pages 2685-2692, Trondheim, Norway, May 2009.

A. Tusek, C. L. Müller, J. Supper, A. Zell, Z. Kurtanjek, and I. F. Sbalzarini. **Systems biology markup language: Case study of T-cell signal transduction network.** In Proceedings of the 29th International Conference on Information Technology Interfaces (ITI07), pages 651-656. IEEE, 2007.

R. Dölling, H. Mielenz, and C.L. Müller. **Efficient Simulation of Automotive Multi-Physical Systems by Optimization with Evolutionary Algorithms**, Applied Simulation and Modelling, Mallorca, 2007.

## Other Scientific Publications

C. L. Müller. **Exploring the common concepts of adaptive MCMC and Covariance Matrix Adaptation scheme.** In: Theory of Evolutionary Algorithms, number 10361 of Dagstuhl Seminar Proceedings, Schloss Dagstuhl, Leibnitz-Zentrum für Informatik, Germany, 2010.

C. L. Müller, G. Paul, and I. F. Sbalzarini. **Sensitivities for free: CMA-ES based sensitivity analysis.** In: Abstracts of the 5th International Conference on Sensitivity Analysis of Model Output (SAMO), Budapest, Hungary, pages 123-124, 2007.

## Theses

C. L. Müller. **Parameter Sensitivity Analysis in Behavioral and Analog Circuit Simulations of Neuro-Fuzzy Models.** Diploma thesis, Robert Bosch GmbH and Department of Information and Cognitive Sciences, University of Tübingen, 2006.

C. L. Müller. **Die Erde dreht sich zu laut – Gedichte von der schwedischen Oberfläche.** Certificate thesis, Studio Literatur und Theater, University of Tübingen, 2005.

C. L. Müller. **High order accurate numerical solution of the linearized Euler equations for sound propagation in the atmosphere.** Master thesis, Department of Computer Science, University of Uppsala, 2004.

# Curriculum Vitae

## Christian L. Müller

| | |
|---|---|
| Date of birth: | April 14, 1979 |
| Place of birth: | Schesslitz, Germany |
| Citizenship: | Germany |

**Education**

| | |
|---|---|
| 2006 – 2010 | PhD student at the Institute of Theoretical Computer Science<br>ETH Zürich, Switzerland, and Swiss Institute of Bioinformatics<br>Advisor: Prof. Dr. Ivo F. Sbalzarini<br>Academic title: Dr. sc. |
| 2005 – 2006 | Studies at University of Tübingen, Tübingen, Germany<br>Major: Bioinformatics<br>Certificate: Literature & Poetry<br>Academic title: Dipl. Bioinf./Inf. |
| 2003 – 2004 | Studies at Uppsala Universitet, Uppsala, Sweden<br>Major: Computational Science<br>Academic title: M. Sc. (Computer Science) |
| 1999 – 2002 | Studies at University of Tübingen, Tübingen, Germany<br>Major: Bioinformatics<br>Degree: Pre-diploma Bioinf. |
| 1989 – 1998 | High school at Franz-Ludwug Gymnasium, Bamberg, Germany<br>Degree: Abitur (German high school diploma) |
| 1986 – 1989 | Primary school in Schesslitz, Germany |

**International experience**

| | |
|---|---|
| July 2007 – Oct 2007 | Visiting scientist at Mediterranean Institute for Life Science<br>Split, Croatia |
| Sept 2003 – Dez 2003 | Maitre d'Hotel (Hovmästare), Södermanland Nerike Nation<br>Uppsala, Sweden |

**Work experience**

| | |
|---|---|
| 2005 – 2006 | Internship Automotive Electronics & Diploma thesis<br>Robert Bosch GmbH, Reutlingen, Germany |
| 1998 – 1999 | Civil service, mobile social service<br>Malteser, Bamberg, Germany |

**Scholarships, Awards, and Grants**

| | |
|---|---|
| 2010 | Member of the EU FP7-ICT Grant consortium<br>on Computational Geometric Learning<br>Coordinator: Prof. Dr. Joachim Giesen, University of Jena<br>ETH Zürich contact person: Dr. Bernd Gärtner |
| 2010 | Best Paper Award, EvoNum, 2010 |
| 2009 | ACM Travel Grant, GECCO, 2009 |
| 1999 – 2005 | Stipend according to the<br>Bayerischen Begabtenförderungsgesetz für besonders Begabte<br>(Bavarian study assistance for gifted pupils) |
| 1998 | Top of the class of 1998 at the Franz-Ludwig Gymnasium |

**Languages**

| | |
|---|---|
| German | Native |
| Swedish | Fluent |
| English | Fluent |
| Latin | Basic |