Large Deviations of Convex Hulls of Random Walks and Other Stochastic Models

Von der Fakultät für Mathematik und Naturwissenschaften der Carl von Ossietzky Universität Oldenburg zur Erlangung des Grades und Titels eines

Doktors der Naturwissenschaften, Dr. rer. nat.

angenommene Dissertation

von Herrn **Hendrik Schawe**
geboren am 17.07.1991 in Oldenburg

Gutachter:	**Professor Dr. Alexander K. Hartmann**
Weiterer Gutachter:	**Professor Dr. Joachim Krug**
Tag der Disputation:	**19.03.2019**

Summary

In the thesis at hand Monte Carlo methods originating from statistical physics are applied to study various problems in far more detail than before. While all those problems have in common that they were up to now mainly studied in regards to the mean values of some observable, in this thesis the full distribution including very rare events with probabilities in the order of 10^{-100} and smaller are obtained and discussed.

The first and largest project of this thesis is about the distribution of the volume and surface of the convex hulls around the traces of random walks. The first part of this project looks at the hulls of standard random walks. For this rather simple model much progress was made in the last decades and it is the only problem of this thesis for which prior numerical results of the whole distribution exists in the special case of two dimensions. Therefore this thesis focuses on a generalization to higher dimensions. The second part of this project scrutinizes more complicated types of random walks which interact with their past trajectory. This interaction makes these random walks suitable as models for, e.g., polymers. The same interaction also leads to an increased difficulty in obtaining results analytically, such that the numerical examination of the whole distribution seems worthwhile.

The second project examines the dis-

Zusammenfassung

In dieser Dissertation werden Monte-Carlo-Methoden, die aus der statistischen Physik stammen, genutzt, um unterschiedliche Probleme genauer zu untersuchen als bisher üblich. Alle hier untersuchten Probleme wurden bisher hauptsächlich im Hinblick auf Mittelwerte von bestimmten Messgrößen untersucht. In dieser Dissertation hingegen wird die gesamte Verteilung inklusive sehr seltener Ereignisse mit Auftrittswahrscheinlichkeiten von weniger als 10^{-100} ermittelt und diskutiert.

Am Anfang dieser Dissertation behandeln wir die Verteilung des Volumens und der Oberfläche von konvexen Hüllen, die von einem Random-Walk besuchte Orte einschließen. Zunächst werden dort die die konvexen Hüllen von einfachen Random-Walks untersucht. Für dieses Modell wurden in den letzten Jahrzehnten bereits einige Fortschritte erzielt und es ist das einzige Modell dieser Dissertation für das bereits numerische Ergebnisse für den zweidimensionalen Spezialfall über die Verteilung bekannt sind. Dieser Teil untersucht die entsprechende Generalisierung für höhere Dimensionen. Der zweite Teil dieses Projekts erforscht kompliziertere Random-Walk-Modelle, die mit der von ihnen hinterlassenen Spur wechselwirken. Diese Wechselwirkung macht sie zu geeigneten Modellen beispielsweise für Polymere. Die Wechselwirkung mit tribution of the ground-state energy of a generalized random-energy model, a toy model from statistical physics with applications to phase transitions and spin glasses. There we find a universal asymptotic form for the distribution of the ground-state energies in the limit of large systems, only dependent via two parameters on the behavior of the underlying distribution of the single energy levels in the system.

The third project scrutinizes the distribution of the length of the longest increasing subsequence of different types of random sequences. This very simple model is connected to statistical physics via its relation to the Kardar-Parisi-Zhang universality class, which describes the fluctuations of the surface of many growth processes. For a case with known asymptotic distribution of the length we can show a convergence of our measured distributions to the asymptotic form for very large parts of the distribution. For another case we can confirm a proposed scaling law also in the far tails of the distribution.

The fourth project of this thesis takes a look at the robustness of networks. Since all systems of interacting objects, be it social networks, energy grids or theoretical models on grids or more complicated topologies, can be modeled with networks, it is of fundamental interest how robust these systems are to failures of single objects. Therefore we looked at a rather simple property of networks, the size of the largest biconnected component. The biconnected component is invulnerable to failures of one single object, such that a large biconnected component is an indication for a robust network. We studied the distribution of its size for two otherwise very well studied

sich selbst führt allerdings zu Schwierigkeiten, analytische Ergebnisse zu erhalten, sodass die numerische Erforschung der Verteilung lohnend scheint.

Das zweite Projekt untersucht die Verteilung der Grundzustandsenergien eines verallgemeinerten Random-Energy-Modells, ein stark vereinfachtes Modell der statistischen Physik zur Untersuchung von Phasenübergängen und Spingläsern. Für die Verteilung der Grundzustandsenergien finden wir eine universelle asymptotische Form im Grenzfall von großen Systemen, die nur durch zwei Parameter vom Verhalten der Verteilung abhängt, aus der die einzelnen Energiestufen gezogen werden.

Das dritte Projekt ermittelt die Verteilung der Länge der längsten aufsteigenden Teilfolge von unterschiedlichen Typen von zufälligen Folgen. Dieses einfache Modell ist von Interesse für die statistische Physik, da es eng mit der Kardar-Parisi-Zhang Universalitätsklasse zusammenhängt, die die Fluktuationen der Oberfläche bei vielen Wachstumsprozessen beschreibt. Für eine Variante des Modells beobachten wir die Konvergenz gegen die analytisch bekannte asymptotische Form. Für eine andere Variante bestätigen wir eine zuvor vorgeschlagene Form in den Bereichen der Verteilung, die extrem unwahrscheinliche Ereignisse beschreiben.

Das vierte Projekt dieser Dissertation erforscht die Stabilität von Netzwerken. Da alle Systeme, die aus wechselwirkenden Objekten bestehen, seien es soziale Netzwerke, Stromtrassen oder theoretische Modelle, die auf Gittern oder komplizierteren Topologien definiert sind, durch Netzwerke modelliert werden können, ist die Stabilität bei Ausfall von einzelnen Objeknetwork models.

ten von grundlegendem Interesse. Deshalb betrachten wir in diesem Projekt eine vergleichsweise einfache Messgröße von Netzwerken: Die Größe der größten Zweifach-Zusammenhangskomponente. Solche Komponenten sind immun gegen den Ausfall eines einzelnen Objekts, sodass ihre Größe ein geeigneter Indikator für die Stabilität eines Netzwerks ist. In dieser Dissertation untersuchen wir die Verteilung dieser Messgröße für zwei bekannte Zufallsgraphmodelle.

Contents

Summary iii

Contents vii

List of Figures ix

1. Introduction 1

2. Large Deviations: Background and Numerical Methods 3
- 2.1. Large Deviation Theory 3
- 2.2. Monte Carlo Methods 8
  - 2.2.1. Simple Sampling 9
  - 2.2.2. Variance Reduction Techniques 9
  - 2.2.3. Markov chain Monte Carlo: The Metropolis Algorithm 15
  - 2.2.4. Metropolis Algorithm to Sample the Large Deviation Regime 19
  - 2.2.5. Wang Landau Sampling 22

3. Convex Hulls of Random Walks 29
- 3.1. Current State of Research 29
  - 3.1.1. Analytic Approach with Cauchy's Formula 32
- 3.2. Research Question 35
- 3.3. Models and Methods 35
  - 3.3.1. Random Walks 36
  - 3.3.2. Convex Hulls 48
- 3.4. Results 54

4. Groundstate Energy of a Generalized Random Energy Model 57
- 4.1. Current State of Research 57
- 4.2. Research Question 60
- 4.3. Models and Methods 60
- 4.4. Results 63

5. The Longest Increasing Subsequence 67
- 5.1. Current State of Research 69
- 5.2. Research Question 71
- 5.3. Models and Methods 72
  - 5.3.1. Patience Sort 72
- 5.4. Results 74

6. The Largest Biconnected Component of Random Graphs **77**
6.1. Definitions 79
6.2. Current State of Research 80
6.3. Research Question 83
6.4. Models and Methods 84
6.4.1. Finding Biconnected Components 84
6.4.2. Erdős-Rényi Graphs 85
6.4.3. Barabási-Albert Graphs 85
6.5. Results 87

Bibliography **89**

Index and Glossary **103**

Appendices **107**

A. Publications **109**
A.1. Convex hulls of random walks in higher dimensions: A large-deviation study 110
A.2. Large deviations of convex hulls of self-avoiding random walks 120
A.3. Large Deviations of Convex Hulls of the “True” Self-Avoiding Random Walk 130
A.4. Ground state energy of noninteracting fermions with a random energy spectrum 142
A.5. Large deviations of the length of the longest increasing subsequence of random permutations and random walks 150
A.6. Large-deviation properties of the largest biconnected component for random graphs 158

B. Technical Details **169**
B.1. Gluing the Single Distributions 169
B.2. Further Convex Hull Algorithms 170
B.2.1. Akl-Toussaint Elimination Heuristic 170
B.2.2. Jarvis’ March / Gift Wrapping 171
B.2.3. Chan’s Algorithm 172
B.2.4. Convex Hull Algorithm Implementation for this Study 174
B.3. Generation of Random Numbers 174
B.4. Biconnected Components 176

C. List of Publications and Preprints and Curriculum Vitae **179**

Acknowledgments **183**

List of Figures

2.1. Probability Density Functions and Large Deviation Rate Function for Mean of Binomial Random Variables 5
2.2. Schematic Energy Landscape for Transition Path Sampling 14
2.3. Markov Chain of Random Numbers $\boldsymbol{\xi}$ 18
2.4. Temperature-Based Sampling Scheme 21
2.5. Schedule of the Modified Wang Landau Sampling 26
2.6. Wang Landau Sampling and Temperature-Based Sampling Scheme . . 27

3.1. Example of a Branching Gaussian Walk 31
3.2. Example of Intermittent Behavior Detectable by Local Convex Hulls . 32
3.3. Demonstration of Cauchy's Formula 33
3.4. Examples of Different Random Walk Types 37
3.5. Examples for Two Local Change Moves 38
3.6. Partial Decision Trees for Different Random Walks of Length $T = 5$. 39
3.7. Example of One Pivot Move 40
3.8. Example of a Loop Erasure 42
3.9. Important Sites for the Smart-Kinetic Self-Avoiding Walk 45
3.10. Convex Hulls in Two and Three Dimensions 48
3.11. Andrew's Monotone Chain Algorithm 51
3.12. The Quickhull Algorithm 52
3.13. The Quickhull Algorithm in 3D 53
3.14. Concept of Hypervolume Calculation 54
3.15. Example Scaling of a SAW's Perimeter Distribution 55

4.1. Demonstrating the Central Limit Theorem and Extreme Value Theory 58
4.2. Distribution of the Sum S of the Largest K Values of N Exponentially Distributed Random Numbers 59
4.3. Visualization of a Groundstate of our Random Energy Model 61
4.4. Data Structure used in our Random-Energy Model 62
4.5. Distribution of the Sum S of the K Smallest Values of N Random Numbers Distributed According to an Erlang Distribution 65

5.1. Examples for LIS in Random Permutations and Random Walks 67
5.2. Example of the Ballistic Deposition Model Mappable to LIS 69
5.3. Example for Patience Sort 73
5.4. Example for Backpointer Extension of Patience Sort 73
5.5. Example for the Complete Backpointer Extension of Patience Sort . . 74

5.6. Comparison of the Length of the LIS of Random Permutations with their Known Rate Functions 75

6.1. Examples of Random Graphs 77
  (a) Erdős-Rényi 77
  (b) Watts-Strogatz 77
  (c) Barabási-Albert 77
6.2. Scale-Free Networks 79
  (a) Twitter 79
  (b) Citeseer 79
  (c) Linux 79
  (d) Mouse Genome 79
6.3. Annotated Graph to find Biconnected Components 84
6.4. Examples of Erdős-Rényi Graphs 85
6.5. Examples of Barabási-Albert Graphs 86
6.6. Comparison of the Biconnected Component's Distribution for Different Sizes and with 2-Core 87

B.1. Akl-Toussaint Elimination Heuristic 171
  (a) with quadrilateral 171
  (b) with octagon 171
B.2. Jarvis' March Algorithm 172
B.3. Chan's Algorithm 173
B.4. Benchmarks of Convex Hull Algorithms 174
B.5. Algorithm to Find Biconnected Components 177

1. Introduction

Simulations are the youngest sibling of experiments and theory in physics.[1] One recognizes their relationship easily considering that simulations, similar to theory, are able to scrutinize arbitrary models – realizable or not. Similar to experiments, simulations gather data to derive conclusions about the models. For many theoretically well researched models experimental data is scarce or non-existent and instead simulations are playing the part of experiments in the scientific process. Some real processes are characterized by complex interactions and hence are not suited for analytical treatment, such that simulations are needed to design a theory predicting the results of experiments [1]. Where theory has to approximate too much and where experiments struggle to be designed such that the effects of interest can be observed without drowning in noise, simulations can shine. The perfect control over every aspect of the model and the relative ease to change or disable certain mechanisms of the model make them a capable tool to understand the mechanisms which lead to the behavior of interest in a system. Due to the undeniable increase in computing capability over the last decades, the importance of this complementary branch of physics only increases, as the study of more complex models and harder to observe measurables becomes feasible.

In this dissertation the focus is mainly on intriguingly simple models which my coauthors and I studied, mainly using simulations, during the last years. The studied models originate from diverse fields ranging from statistical physics over combinatorics to graph theory, such that it is most sensible to introduce each in their own section in the course of this thesis. The common theme, which is present in every publication belonging to this thesis, is that for each model we study the behavior of one or two observables of interest in very high detail. For the observable of interest we obtain a large part of their distributions numerically. Especially the far tail behavior, with probabilities far smaller than 10^{-100}, is probed for the first time for the corresponding models. Properties of distributions including these extreme tails, are usually called *large-deviation* properties. Knowledge of the large-deviation behavior enables us to observe properties which are hidden in the part of the distribution inaccessible by conventional sampling methods leading to a deeper understanding of the atypical and rare events. Also it allows us to test, e.g., scaling assumptions over almost the whole range of possible values, which strengthens the confidence in analytical results which are otherwise seldomly tested in these limits.

More background of large deviations and methods which enable the simulational study of large deviations, underlying every publication of this thesis, are introduced first in Chapter 2. It will establish a bit of background of large deviation theory

---

[1] This metaphor is inspired by David Landau's triangle [1, p. 5].

from a mathematical perspective and introduce in detail numerical methods to study large-deviation properties. Due to its centrality to each publication of this thesis, this methodical part constitutes a large portion.

The problems, to which these methods are applied, are formulated in the subsequent Chapters 3 to 6. They are structured in a rather strict way, where for each the *state of the current research* is stated and elaborated by summarizing important milestones in their history in a more throughout fashion than possible in a research article. Directly after establishing the background of the problem the *research question* for the corresponding study will be stated. These sections may contain some vocabulary which might be unfamiliar to the reader. While most concepts should be explained in enough detail to understand the text, some readers might prefer to read the more throughout and formal definitions given in the *models and methods* sections first. In these sections models and methods special to the problem at hand will be explained. Some technical details which would take too much space in the main text are moved to Appendix B; at the corresponding places this appendix will be referenced. The most important results obtained and published by my coauthors and me are very shortly summarized in the concluding *results* section of every problem.

However, for a comprehensive understanding of the results it is recommended to read the research papers in Appendix A, where for every publication also a concise statement of the contributions of every coauthor involved is given. This part of the cumulative thesis at hand consists of 6 manuscripts, of which 5 (Articles A.1, A.2 and A.4 to A.6) are published in peer reviewed journals and the remaining article (Article A.3) is accepted by a peer reviewed journal. All manuscripts in which I am listed as coauthor and which are published or submitted at the time of writing, also to topics not part of this thesis, are listed in Appendix C.

The most central group of models under scrutiny in this thesis is introduced in Chapter 3. There the relevance of convex hulls around random walks, their constructions and the definitions of different random walk models are explained in detail for all models under scrutiny in Articles A.1 to A.3. Chapter 4 will take a look at the distribution of the ground-state energy of a random-energy toy model and which connection to statistical physics and even fundamental stochastics exist. Chapter 5 will scrutinize the distribution of the length of the longest increasing subsequence of different random sequences, which is a simple combinatorial problem with links to growth processes of the Kardar-Parisi-Zhang universality and the Tracy-Widom distribution. At last Chapter 6 motivates a study we conducted for the distribution of the size of the largest biconnected component of random graphs, that is the largest subgraph which stays connected after removing any node.

2. Large Deviations: Background and Numerical Methods

This chapter introduces the reader to the concepts of large deviations and the methods used to study them. These concepts and methods are used throughout all publications belonging to this thesis.

In Section 2.1 we will explore the basic meaning of large deviation theory, though without mathematical rigor or the derivation of analytical tools. This first part should demonstrate the background and an analytical approach to the problems we will handle numerically in this thesis. It should elucidate the reader what a *rate function* is and why we are interested in it.

Consequently Section 2.2 will focus on numerical methods to examine large deviations. It will give a short overview over the historic evolution, starting at the basic concept of Monte Carlo over some historically significant approaches to probe the behavior of specific physical problems efficiently to the methods used in the publications belonging to this thesis, which are described in more detail.

2.1. Large Deviation Theory

Large deviation theory is a mathematically rigorous theory to describe the fluctuations of stochastic processes. This includes the small fluctuations close to the typical values as well as the *large fluctuations* far away from typical values, from which its name stems. Historically [2], large deviation theory as a unified general framework came up in the 1960s and 1970 by the works of Donsker and Varadhan. Although first results can be attributed [3] to Boltzmann, who derived some as the foundation for statistical mechanics, and many other mathematicians anticipating parts of this theory. For example, Cramér [4] found in the 1930s the *large deviation principle* (cf. Equation (2.1)) for the empirical mean, which we will use in the following as an example (cf. Equation (2.2)).

Large deviation theory is mainly a concept used in the context of statistical mechanics, but since statistical mechanics is known to apply its methods to problems of other branches of physics, large deviation theory is applied to a varied spectrum of problems [3]. As a concrete example, take a very recent paper, where it was applied to a problem from fluiddynamics. A discrete model of Taylor dispersion was studied in Reference [5], i.e., given a Poiseuille flow, which is a laminar flow in a cylinder, what is the distribution of the displacements of particles floating in the fluid.[1] While

---

[1] Not to be confused with the flow profile, which is a parabola and known for much longer.

it was known for a long time to be approximately a Gaussian [6], using large deviation techniques lead to corrections to this behavior.

In the remainder of this section large deviation theory will be introduced in rough strokes. Therefore, I will follow the structure of the excellent review article of Hugo Touchette[2] [2] and borrow some notation and examples from it. This introduction to large deviation theory will not contain any proofs, rather it should give an understanding of its usefulness, which will be illustrated with simple examples of some of its core machinery in action. Some of the examples might seem overly technical and may be skipped without compromising the understanding of following chapters. However, they are included in this thesis to give a glimpse of the background and other approaches to the problems, which are a central part of the thesis.

The central prerequisite to apply large deviation theory is that the large deviation principle holds. The large deviation principle fundamentally means that the tail of the distribution decays as an exponential in some parameter n. To be more precise, the probability density function $P(x = x_0)$ needs to behave like

$$P_n(x) = \exp\left(-n\Phi(x) + o(n)\right) \approx \exp\left(-n\Phi(x)\right), \tag{2.1}$$

with the *Landau symbol* $o(n)$ representing terms of order less than n. Φ is called the *rate function* and is the central quantity of large deviation theory. If such a rate function exists, the large deviation principle holds. In the limit of large n, the whole distribution is characterized by the rate function. The parameter n can be anything, a time, a size or a number of iterations. Large deviation theory can only make statements for large values of n, which makes it a perfect fit for statistical physics, where we are mostly interested in the thermodynamic limit, i.e., in systems with $n \to \infty$ elements, such that large deviation theory finds application here. Of course, not every process fulfills the large deviation principle: The probability density function could decay sub- or super-exponentially, or could be too singular. Therefore, in best mathematical fashion, an important part of the problem is always to show the existence of the rate function. This is also addressed in most of the publications of this thesis in Appendix A, but with non-rigorous numerical arguments.

As a simple example of a process fulfilling the large deviation principle, we will look at the mean of n binomial random variables. For historic reasons, which will be laid out in Section 2.2, we look at the random process of winning a game of solitaire. Each game of solitaire is winnable with probability p. We are interested in the distribution of the winrate after n games played. Since every new game of solitaire is played with a freshly shuffled deck of cards, we consider n independent random variables X_i, each is 1 with probability p and 0 with $1-p$. Their mean value

$$A_n = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{2.2}$$

---

[2] During the summer school “Fundamental Problems of Statistical Physics XIV” 2017 in Bruneck, I had the opportunity to attend a lecture of Hugo Touchette about this exact topic [7].

is the random variable describing the winrate. For this toy example the distribution is known to be binomial and the probability density function to be

$$P(A_n = k/n =: a) = \binom{n}{k} p^k (1-p)^{n-k} = \frac{n!}{k!(n-k)!} p^k (1-p)^{n-k}. \qquad (2.3)$$

The parameter n is the number of games played and since large deviation theory does only make statements for large values of n, we can use Stirling's formula $n! \approx n^n \mathrm{e}^{-n}$. Looking at the logarithm of the combinatorial term, yields

$$\ln \frac{n!}{(an)!(n-an)!} = n\left[-a \ln a - (1-a)\ln(1-a)\right].$$

Which leads with the second part of Equation (2.3) to an asymptotic probability density function of

$$\ln P_n(A_n = a) = n\left[a \ln \frac{p}{a} + (1-a) \ln \frac{1-p}{1-a}\right].$$

Comparing this to the expression for the large deviation principle Equation (2.1) one can read off the rate function

$$\Phi(a) = -a \ln \frac{p}{a} - (1-a) \ln \frac{1-p}{1-a}. \qquad (2.4)$$

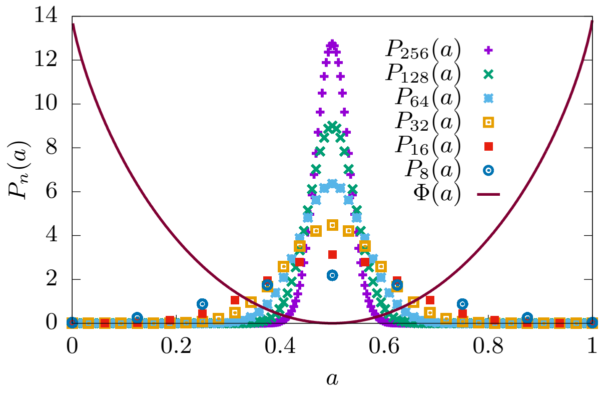


Figure 2.1.: Distribution P_n of the mean a of n binomial random numbers with its corresponding rate function Φ. (While P_n is a discrete distribution, the normalization here pretends that it is a continuous distribution for clarity.)

The rate function $\Phi(a)$ governs how the probability density decays. In Figure 2.1 the probability density is shown for various values of n and $p = 0.5$ together with the rate function $\Phi(a)$. In this example, it is clear that the probability concentrates at the point $a = p$, since $\Phi(p) = 0$ is the unique root of the rate function, i.e., it is the only point where the probability density does not decay exponentially. Or in a more mathematical formulation $\lim_{n\to\infty} P(A_n = p) = 1$, which is also known as the *law of large numbers*.

There is a class of well studied problems, whose rate function Φ has a unique minimum and zero at a_0, is convex and twice differentiable, like the example in Figure 2.1. This kind of rate functions can be obtained via the *Gärtner-Ellis theorem*, which will be introduced below. For these we can approximate the rate function using a Taylor expansion to second order:

$$\Phi(a) \approx \frac{1}{2}\frac{\mathrm{d}^2\Phi}{\mathrm{d}a^2}(a_0)\,(a-a_0)^2\,. \tag{2.5}$$

The corresponding probability density is

$$P(a) = \mathrm{e}^{-n\Phi} \approx \exp\left(-n\frac{\mathrm{d}^2\Phi}{\mathrm{d}a^2}(a_0)\,(a-a_0)^2\right).$$

That means that small fluctuations, near the most probable value a_0 can be approximated by a Gaussian for large n. This is the same statement as the *central limit theorem* would give for our example. This is no coincidence and there is a rigorous proof that the central limit theorem follows from the large deviation principle [8].

The last paragraphs showed some statements about small deviations, which can be derived if the rate function is known. Since the rate function contains the information of the distribution in leading order, i.e., the complete behavior of the distribution in the large n limit, this is not really surprising. The real appeal of large deviation theory is the apparatus of methods to show the existence of a rate function and derive it without a way to directly obtain the large n behavior of the probability density function. We will revisit the example above, for which we derived the rate function directly using Stirling's formula, and derive its rate function using a more formal approach.

Gärtner-Ellis Theorem First, let's look at (a simplified version of) the Gärtner-Ellis theorem. For a random variable A_n the *scaled cumulant generating function* is defined as

$$\lambda(k) = \lim_{n\to\infty}\frac{1}{n}\ln\left\langle \mathrm{e}^{nkA_n}\right\rangle \tag{2.6}$$

with $k \in \mathbb{R}$ and

$$\left\langle \mathrm{e}^{nkA_n}\right\rangle = \int \mathrm{d}a\; \mathrm{e}^{nka} P(A_n = a), \tag{2.7}$$

which is called *moment generating function* and is technically very closely related to a *Laplace transform*. The Gärtner-Ellis theorem states that A_n fulfills the large deviation principle if $\lambda(k)$ is differentiable for all k and in that case the rate function is given by the *Legendre transform* $\Phi(a) = \sup_{k\in\mathbb{R}}\left\{ka - \lambda(k)\right\}$. Technically, this is a Legendre-Fenchel transform, a generalized Legendre transform for non-convex functions, which chooses the supremum[3] of all non-unique branches. This can intuitively be understood as taking the most probable event causing this fluctuation, since this will dominate the behavior of the rate function. This is also known as *Laplace's approximation.* In the case of a unique inversion $k(a)$, this reduces to the classical Legendre transform. The Legendre transform constructs dual functions in the sense that the function's Φ and λ first derivatives are inverses of each other, i.e., $\lambda'(k) = a$ and $\Phi'(a) = k$.

To illustrate this, consider the binomial process from above. The probability density function of the i-th event is $P(X_i = x_i) = p\delta(x_i - 1) + (1 - p)\delta(x_i)$ with the Dirac delta. Now, we want to obtain the distribution of the winrate Equation (2.2) for n events. So looking at the logarithm of the moment generating function Equation (2.7), the *cumulant generating function*, we get

$$\begin{aligned} \ln\left\langle \mathrm{e}^{nkA_n}\right\rangle &= \ln\left\langle \mathrm{e}^{nk\frac{1}{n}\sum_{i=1}^{n} X_i}\right\rangle \\ &= \sum_{i=1}^{n} \ln\left\langle \mathrm{e}^{kX_i}\right\rangle = n\ln\left\langle \mathrm{e}^{kX_1}\right\rangle \\ &= n\ln\left(p\mathrm{e}^k + 1 - p\right), \end{aligned}$$

where we use that every event is identically distributed and independent and evaluate the integral $\langle\cdot\rangle$ over the delta functions. This expression can be inserted into Equation (2.6)

$$\begin{aligned} \lambda(k) &= \lim_{n\to\infty} \frac{1}{n} n\ln\left(p\mathrm{e}^k + 1 - p\right) \\ &= \ln\left(p\mathrm{e}^k + 1 - p\right). \end{aligned}$$

To perform the Legendre transform as the last step, we first need to perform the inversion

$$\begin{aligned} a &= \lambda'(k) \\ &= \frac{p\mathrm{e}^k}{p\mathrm{e}^k + 1 - p} \\ \Rightarrow \quad k(a) &= \ln\frac{a(p-1)}{p(a-1)}. \end{aligned}$$

With these two intermediate results, the evaluation of the Legendre transform reduces to just some elemental algebra

$$\begin{aligned} \Phi(a) &= ka - \lambda(k) \\ &= a\ln\frac{a}{p} + (a-1)\ln\frac{p-1}{a-1}, \end{aligned}$$

which is identical to Equation (2.4).

---

[3] The supremum is the smallest upper bound, so for closed sets it is equivalent to the maximum.

Note however that this method will not always work. For example, rate functions which are not strictly convex can not be obtained with this method. While there are more analytical tools, which might work in those cases, it is certainly out of scope of this thesis to give a comprehensive overview over the analytical methods of large deviation theory. The thesis at hand instead uses a toolset of sophisticated numerical sampling methods to obtain estimates of the distribution and estimate the rate function from these data. The major ideas needed for those numerical methods, historical milestones and detailed descriptions of the algorithms used to generate the published results of this thesis are presented in Section 2.2.

2.2. Monte Carlo Methods

The term *Monte Carlo methods* refers to a large class of methods which use randomness to calculate some quantity of interest. A famous Monte Carlo algorithm for the calculation of π is *Buffon's needle*. This was originally posed in the 18th century as a geometrical riddle asking for the probability of a needle of length l thrown on a piece of paper with parallel lines of distance t to intersect a line. For short needles $l < t$, the probability is[4] $p = \frac{2l}{\pi t}$. Performing an experiment to throw N needles and count the number x of needles crossing a line leads to an estimate for $p \approx x/N$, which can be used for an estimate of $\pi \approx \frac{2lN}{xt}$. The estimate is subject to a statistical error scaling as $\mathcal{O}(N^{-1/2})$, which makes this method a particular bad algorithm for estimating π.

The founding myth [9] of modern Monte Carlo goes back to 1946, when Stanisław Ulam, recovering from a disease, played *Canfield Solitaire*, a variant of solitaire with a low winning probability. The small number of games he could win probably prompted him to think about the probability that a given game is winnable. After futile attempts using combinatorial arguments, he wondered if it would be more practical to just play it a hundred times and observe how often he would win. A task he assumed the (by some definitions first) programmable electronic computer ENIAC (Electronic Numerical Integrator and Computer, 1945) at his working place, the Los Alamos Laboratories, could do. While it is not delivered whether he ever used the ENIAC for this problem, we have to assume that compute time was too valuable at the time. But he thought about applying this idea to physical problems, namely neutron diffusion. Later, together with John von Neumann, they planned the first Monte Carlo simulations for physical problems to be run on the ENIAC. Around this time also the name "Monte Carlo" was coined. It was suggested[5] by Nicholas Metropolis as a codename

---

[4] A classical way to derive this result (as featured in the Wikipedia), is to split the probability p into p_1 and p_2. Each needle is defined by two independent random variables, the distance of its center to the next line x and its acute angle with the line θ. Since both are uniformly distributed, their probability density functions are $p_1 = 2/t$ if $0 \le x \le t/2$ and $p_2 = 2/\pi$ if $0 \le \theta \le \pi/2$. A needle crosses a line, if the distance to the line is smaller than the projection of the needle perpendicular to the line; to be precise if $x < l/2\cos(\theta)$. Integrating the joint probability in these borders yields $\int_0^{\pi/2} \mathrm{d}\theta \int_0^{l/2\cos(\theta)} \mathrm{d}x \frac{4}{t\pi} = \frac{2l}{t\pi}$.

[5] "not unrelated to the fact that Stan had an uncle who would borrow money from relatives because he 'just had to go to Monte Carlo'." in the words of Metropolis himself [10]

for the project.[6]

2.2.1. Simple Sampling

To stay at the example of a game of solitaire, the simple method envisioned by Ulam is to shuffle a deck of cards, play out the game of solitaire and note if the game was won or not. If we iterate this N times, we can generate a histogram of our data to approximate a probability density function. We could even collect some more complicated observables, say the number of cards we have to touch. The downside is that only realizations from the peak of the probability density functions are sampled, i.e., to reach a configuration which occurs with probability p, roughly $1/p$ samples need to be generated. Even worse, every approximation of an expected value is tainted with a statistical error of order $\mathcal{O}(N^{-1/2})$. Therefore it is infeasible to obtain the large deviation tails of the probability density function for probabilities of $p \lesssim 10^{-10}$ with this method.

Simple sampling will always yield more samples from the high probability region of a distribution, but those samples are useless to get information about the tails of the distribution. This is not only a problem when the region of interest is in the far tail of a distribution, but sometimes realizations are not easy to generate uniformly. For example look at the self-avoiding random walk (a more detailed elaboration of this model and this problem is in Section 3.3.1), the ensemble of all random walks on a square lattice, which do not visit any site twice. To perform simple sampling, one has to generate a random walk and discard every realization which visits any site twice. Since the chance to step on an already visited site is roughly constant per step, this leads to exponential *attrition* in the length of the walk, i.e., to generate one self-avoiding random walk of length T we would need to create $\mathcal{O}(\exp(T))$ attempts. This is not feasible for any but the smallest systems. So we would rather concentrate on the important realizations which do not step twice on any site.

Due to this drawback of simple sampling, we need a better way to sample the more important parts of the distribution of interest. Which part is more important is, of course, dependent on the question we have. In the next sections, some techniques are presented to collect samples more cleverly suited to the problem at hand. This part culminates in the methods that were used in the studies which are part of this thesis.

2.2.2. Variance Reduction Techniques

Approaches to improve simple sampling by reducing the statistical error are almost as old as Monte Carlo itself. A lower error means that fewer samples are needed, which results in less compute time and enables therefore the study of larger or more complex models. These techniques were collected under the term *variance reduction* [11].

Probably the most important technique, and the technique used in the publications of this thesis, is *importance sampling*. Before we look at importance sampling some

---

[6] A codename was required since the work in the Los Alamos Laboratories was for military purposes – at this time fission and the first fusion weapons – and considered secret.

other interesting approaches are introduced. This is a non-exhaustive historical and methodical overview over the field. Also some of the presented ideas can be combined.

We change our example from a game of solitaire to something simple to simulate and more suited for a casino in Monte Carlo: Rolling dice (cf. Reference [11]). As a toy example, we want to know the probability p for two six-sided dice to sum to 3, i.e., tossing ⚀ ⚁ or ⚁ ⚀. The relative error[7] for a sample of N tosses of this binomial process is

$$\Delta = \sqrt{\frac{1-p}{Np}}. \tag{2.8}$$

As an alternative to increasing the sample size N to lower the relative error, it is possible to modify the generation of the samples such that the effective value of p is larger. We explore some ideas in this section and test them on this very simple dice rolling example. Also we look for each of the techniques at a real application to demonstrate its capabilities beyond a toy model.

For the toy model however, a comparison of results for $N = 10^4$ dice tosses is visualized in Table 2.1. The estimate of the standard error is obtained using bootstrap resampling [12–14]. Since this comparison is in no way rigorous and the choice of the algorithm details under the concepts is not optimal, it is more meant as a demonstration that the methods work in principle and is not to be mistaken as a quality assessment.

exact	$0.055\overline{5}$
simple sampling	0.0544(23)
Russian roulette & splitting	0.0569(17)
importance sampling	0.0549(11)

Table 2.1.: Estimated probability of two dice summing to 3. Error estimates via bootstrap resampling. Data gathered over $N = 10^4$ samples each.

Russian Roulette & Splitting is a catchy name[8] for a simple idea. If we have a set of samples s_i with some sensible weights w_i we assigned, e.g., for each of N samples obtained by simple sampling $w_i = 1/N$, we can calculate a mean value as $\langle a \rangle = \sum_{i=1}^{N} w_i s_i / \sum_{i=1}^{N} w_i$. Without changing the result, we can *split* (that means duplicate) a sample s_j with weight w of this set into n samples with arbitrary weights w_i, such that $w = \sum_{i=1}^{n} w_i$.

Doing this in the end, after all samples are gathered, does not have any benefit, of course. But if the creation of our samples consists of multiple steps, we might be able to judge which partial realization is important and which is not. We can then

---

[7] The standard error of a sample is $s = \frac{\sigma}{\sqrt{N}}$, where the standard deviation for a binomial process is $\sigma = \sqrt{p(1-p)}$. The relative error is therefore $\Delta = s/\mu$ with the mean $\mu = p$.

[8] According to Reference [11] it was named by von Neumann and Ulam.

split the important ones into m partial realizations with each $1/m$ of the previous weight and "branch" out from this point to create more samples, which are important. Similarly, the *Russian Roulette* part of the name hints at the inverse procedure. If we can identify partial realizations which will probably lead to unimportant samples, we can discard them with probability[9] q and increase the weight of the surviving partial realizations to $1/(1-q)$ times their previous weight.

For our example, one could toss one die first. If it shows ⚀ or ⚁, we deem this partial realization as important. We will perform the second toss twice and count both results each with half the weight. If we toss one of ⚂ ⚃ ⚄ ⚅ with the first die, we see that we can not reach a sum of three anymore and deem this partial realization as unimportant. With a probability of 2/3, we will not toss the second die, otherwise, we will count this result with three times the weight.

Pruned-Enriched Rosenbluth Method (PERM) as introduced in Reference [15] uses the basic idea of the above introduced "Russian Roulette & Splitting" and refines it for the application to the self-avoiding random walk, which is a central topic of this thesis. While the intricacies of generating uniformly distributed instances of self-avoiding random walks will be explained in more detail in Section 3.3.1, it is not sufficient to grow a walk and avoid already occupied sites. If this strategy is pursued, every time we avoid an already visited site, we need to adjust the weight of this configuration. Consider we are at the n-th step. There are m_n allowed, i.e., unoccupied, sites for the next step of M possible sites. First we look at the extreme cases: If $m_n = 0$, no step is possible and the whole configuration will be weighted with 0. If all sites are unoccupied we have a chance of $m_n/M = 1$ to choose an allowed site for the next state, such that the weight of this configuration stays the same. For values of m_n in between the next random step will lead with probability m_n/M on an allowed site, otherwise the walk may not be counted. So, if we ensure that the next step is on an unoccupied site, e.g., by drawing random numbers until the corresponding step is allowed, we have to adjust the weight of the resulting configuration by a factor to m_n/M. For the whole walk of T steps (after multiplying the weight with M^T, which does not change the result, to arrive at the same expression as Reference [15]) we arrive at a weight of our configuration of $W_T \propto \prod_{n=1}^{T} m_n$. This technique is named Rosenbluth-Rosenbluth method [16] after the inventors. While this is much better than simple sampling, i.e., starting from scratch after stepping on an occupied site, at large lengths T it typically leads to single samples with very large weights to dominate the statistics.

This is where the *enrichment* process comes into play. Enrichment [17] is basically another word for "splitting" as introduced above. In PERM the splitting happens as soon as the partial weight W_n exceeds some threshold value. In that case, two realizations with half the weight are grown from this partial realization. The opposite effect of too many irrelevant configurations with very small weights is counteracted by *pruning*, which is a less offensive word for the same idea as "Russian roulette," explained above. As soon as the partial weight W_n of a partial realization is smaller

---

[9] In contrast to the game, q does not have to be 1/6, but can be chosen arbitrarily.

than some threshold, this configuration is discarded with probability 1/2 and otherwise its weight is doubled. This procedure ensures that the weights of all samples are in a predefined range between the thresholds, and therefore contribute all roughly the same to the measured property.

In this method also importance sampling (an explanation will follow below) can be implemented to sample efficiently, e.g., the canonical ensemble, by biasing the site which will be visited next. But since the study on the properties of self-avoiding random walks in Article A.2 does not use PERM, but a different method to generate self-avoiding random walks in an efficient way, we will not go into further detail here.

Importance Sampling generates more important samples and fewer unimportant ones leading to better statistics. Here "important" means again "most useful to the question we want to answer". Like above it can be used to reduce the variance by increasing the effective probability (which needs to be reweighted).

Coming back to our dice – an example that might look quite forced – we do not sample the natural ensemble, i.e., in this case uniform among all realizations. Instead, we will bias our sampling and reweight our results afterwards.

If we want to know the mean probability r of the sum of two dice being 3, which is possible with the events ⚀ ⚁ and ⚁ ⚀, we want to sample these important values more often. Therefore we load the dice, such that the probability to roll a ⚀ or ⚁ is doubled, leading to a non uniform sampling of the events. The rationale why this is preferable is that the probability r' to encounter the event of a sum of 3 in this biased sampling is larger and the relative error Equation (2.8) therefore smaller (cf. Table 2.1). Of course, the biased r' obtained by sampling with loaded dice must be reweighted. Since we know that for a sum of 3 we need to roll ⚀ and ⚁ in arbitrary order, which is four times more probable than before due to the loading, we arrive at $r = r'/4$.

More generally, if we want to obtain an expectation value $\langle S \rangle$ for a function $S(\mu)$ taking realizations μ of the system as an argument, which are distributed according to some probability density function $p(\mu)$, we have to evaluate the integral

$$\langle S \rangle = \int_{\mathcal{C}} \mathrm{d}x \, S(x) p(x) \tag{2.9}$$

over the configuration space $\mathcal{C}$. We can use simple sampling to obtain the mean $\bar{S}$ as an estimate of the expectation value, by drawing uniform samples μ:

$$\bar{S} = \frac{\sum_\mu S(\mu) p(\mu)}{\sum_\mu p(\mu)}. \tag{2.10}$$

Formally, we can multiply Equation (2.9) with $1 = \frac{q(\mu)}{q(\mu)}$, with a distribution $q(\mu)$ of our choice, i.e.,

$$\langle S \rangle = \int_{\mathcal{C}} \mathrm{d}x \, S(x) \frac{p(x)}{q(x)} q(x). \tag{2.11}$$

For clarity we will consistently refer to samples drawn according to q as ν and uniform samples as μ. If we generate samples ν according to $q(\nu)$, the evaluation of the mean becomes

$$\bar{S} = \frac{\sum_\nu S(\nu)p(\nu)/q(\nu)}{\sum_\nu p(\nu)/q(\nu)}, \tag{2.12}$$

where $\frac{p(\nu)}{q(\nu)}$ is called *importance sampling factor*. A clever choice of $q(\nu)$ might lead to a lower variance and therefore a higher precision of $\bar{S}$.

While this is the basic, general idea, there are still some tricky questions to answer. First, it is not necessarily easy to generate realizations ν according to the distribution $q(\nu)$ and Section 2.2.3 will show a solution to this problem.

Second, it is not always clear what $q(\nu)$ should look like. To understand which choice of $q(\nu)$ is sensible, we first have to understand why this method might bring an improvement at all. A criterion for an improvement of an estimate of an expectation value would be a lower variance of the sample. Therefore we have to think about a choice of $q(\nu)$ such that the variance of $S(\nu)p(\nu)/q(\nu)$ is small. The perfect case would of course be that this term is equal to the actual expectation value $\langle S \rangle$ for every sample ν, resulting in a variance of zero. This case is rather academic since it would require the knowledge of $S(\nu)$ and $p(\nu)$ for every ν, in which case we can directly calculate the expectation value without any need for sampling. But maybe we can guess how $S(\nu)p(\nu)$ behaves. For our simple dice example, we know that $p(\nu)$ is uniform for fair dice and $S(\nu) = \delta_{\nu,\text{⚀⚁}} + \delta_{\nu,\text{⚁⚀}}$. What we did above, was a good guess for the form of $S(\nu)$, which approximates the actual form, resulting in a reduced variance.

While mean values are maybe the most common use case for importance sampling, it is far more generally applicable. We are interested in the large deviation behavior of different distributions, i.e., in the very low probability tails. The recipe is the same as above, we have to generate realizations according to a known distribution q in the far tails and can then reweight them according to the chosen distribution q. The exact procedure will be explored in detail in Section 2.2.3.

Transition Path Sampling While the canonical example for importance sampling – at least among computational physicists interested in thermodynamic systems in equilibrium – is the *Metropolis algorithm* applied to the *Ising model* for ferromagnetism [18–20], we will instead look at Transition Path Sampling [21]. The reason for this decision is that the Ising model, also called the fruit fly of statistical mechanics, is probably one of the most well known and most often explained models.[10] Also, *transition path sampling* has the advantage that it is used to study rare events and determine the very small transition rates of those events. This makes it a suitable topic in the context of this thesis.

The transition path sampling method originates in chemistry. As an example [23], take a simulation of a few hundred water molecules and one weak acid molecule. For

---

[10] Also more than once by the author of this thesis [20, 22].

a classical simulation, the timestep has to be in the order of 10^{-18} seconds, since this is the timescale of the fastest motions of the water molecules, e.g., vibrational modes. However, the halftime of the acid before dissociation is in the order of 10^{-3} seconds. If we want to study this dissociation process using a proper molecular dynamics simulation, we would have to perform 10^{15} timesteps to observe $\mathcal{O}(1)$ dissociation event. To get a reasonable estimate of the rate of the dissociation event, we would need to observe at least tens or hundreds of events. This is not feasible with current computers.

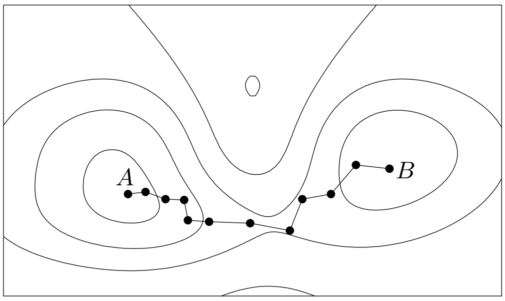


Figure 2.2.: Schematic energy landscape along two arbitrary reaction coordinates, e.g. bond length, bond angles or even non-geometric coordinates. The black lines mark lines of equal energy and the energy difference between two neighboring lines is the same. The points A and B are located in valleys and separated by an energy barrier, such that dynamic trajectories will only seldomly transition. A discrete path generated by the Metropolis algorithm (cf. Section 2.2.3) is shown. Transition path sampling accumulates statistics over these paths to estimate the very small transition rates from A to B.

What transition path sampling does instead is a Monte Carlo simulation of trajectories using the above introduced importance sampling idea and the below introduced Markov chain technique to generate trajectories of rare but important events, e.g., trajectories leading to the above mentioned example of the dissociation of the acid. Consider an energy landscape like Figure 2.2, where in order for an interesting event to happen, a particle needs to travel from A to B. Since there is an energy barrier on every path from A to B, this event is rare in a conventional molecular dynamics simulation. Instead, a discrete path is generated using the Metropolis algorithm (cf. Section 2.2.3). The discrete path is nothing like a trajectory created by a molecular dynamics simulation. Rather, it consists of a number of straight sections, where at some points the (generalized) impulse changes abruptly. The points of the impulse changes are marked by dots in Figure 2.2. Despite the trajectories not being realistic, this method maintains the correct statistics to estimate the transition rates from the samples of the discrete trajectories. Since the energy landscape is not known beforehand and can be very high dimensional, the authors of Reference [23] call it “throwing ropes over rough mountain passes, in the dark.”

The Metropolis algorithm starts with a random discrete trajectory from A to B, changes it a bit by changing the impulse at one of the dots and might revert the change either because the new path does not end in B anymore or to maintain the correct statistics of the visited trajectories. For example, the temperature T plays a large role in chemical processes, since the probability to be at some position in the energy landscape with energy E follows a Boltzmann distribution (in the canonical ensemble) $p \propto e^{-E/k_B T}$. Of course, the probability to take a path up- or downhill in the energy landscape will depend also on the temperature. The exact mechanism how to achieve this is not essential at this place, but will be shown in Section 2.2.3 in general and is described, e.g., in Reference [23] for the transition path sampling case. Of the ensemble of all trajectories from A to B improbable trajectories will be sampled less often than probable trajectories. If the sample is large enough, this allows a precise characterization of the trajectories from A to B, e.g., the rate with which this transition happens.

2.2.3. Markov chain Monte Carlo: The Metropolis Algorithm

The fundamental ingredient for importance sampling is the generation of samples according to a specific distribution. *Markov chains* are an instrument to construct samples distributed according to any distribution. This technique is referred to as *Markov chain Monte Carlo* (*MCMC*). We will spend some time here to introduce Markov chains and a lot of notation used in the following sections. Note that the structure and notation of this section follows largely References [1, 19].

Fundamentally *MCMC* uses a *Markov process* to generate realizations from a specific distribution efficiently. This stochastic process is discrete in time and has at time t some state μ. The *Markov property* means that at any time t_i the state ν only depends on the state μ at t_{i-1}. This means that the time evolution can be characterized by transition probabilities $P(\mu \to \nu)$. For the transition probabilities to be useful for our application, we assume them to be time independent, i.e., only dependent on the current state and the state of the next step. Also the probability that a transition occurs needs to be unity, i.e.,

$$\sum_{\nu} P(\mu \to \nu) = 1, \tag{2.13}$$

where ν does not need to be different from μ. Starting this process in an arbitrary state, will create a timeseries of states, which is commonly called *Markov chain*.

To sample the wanted distribution, one has to ensure that every state can be reached by the Markov chain – one says, the Markov process must be *ergodic*. Therefore the transitions must be designed in a way that they can reach every configuration from any other in finite time.[11]

As mentioned before, the Markov process is a means to the end of generating states μ according to some distribution p_μ. So the Markov process needs to be stationary, i.e.,

---

[11] While this may sound like it is no problem, there are examples for algorithms which do not fulfill *ergodicity* and therefore lead to wrong results. A fitting example is the “slithering snake” (also called “reptation”) move of the self-avoiding random walk [24].

$\frac{\mathrm{d}p_\mu(t)}{\mathrm{d}t} = 0$. To ensure this, we look at the *master equation*, which makes a statement about the time evolution of stochastic processes

$$\frac{\mathrm{d}p_\mu(t)}{\mathrm{d}t} = \sum_\nu \left[p_\nu(t)P(\nu \to \mu) - p_\mu(t)P(\mu \to \nu)\right]. \tag{2.14}$$

It can be seen as a continuity equation for probabilities, since it basically states that the change in probability of the Markov process to be in state μ on the left-hand side is the difference of the rate entering this state and the rate leaving this state. For a stationary state p_μ independent of the time t it is therefore necessary that the rate of the transitions into some state is the same as the rate out of this state

$$\sum_\nu p_\mu P(\mu \to \nu) = \sum_\nu p_\nu P(\nu \to \mu). \tag{2.15}$$

This is called *global balance*. The classical method to fulfill this balance condition, is to require *detailed balance*, which means that the summation goes over the same terms, i.e.,

$$p_\mu P(\mu \to \nu) = p_\nu P(\nu \to \mu). \tag{2.16}$$

It ensures that the transition rate from μ to ν is the same as the rate from ν to μ. Note, however, that there are algorithms using the larger freedom granted by global balance [25].

The transition probabilities need to be chosen properly to sample the wanted distribution. The classical example is the Boltzmann distribution, where at given temperature T the state μ with some assigned energy E_μ should be generated with probability

$$p_\mu = \exp\left(-\frac{E_\mu}{k_B T}\right)/Z, \tag{2.17}$$

where Z is the partition function and the Boltzmann constant $k_B = 1$ in natural units. For two states μ and ν detailed balance Equation (2.16) leads directly to an expression for the transition probabilities

$$\frac{P(\mu \to \nu)}{P(\nu \to \mu)} = \frac{p_\nu}{p_\mu} = \exp\left(-\frac{E_\nu - E_\mu}{T}\right). \tag{2.18}$$

From this we can design the transition probabilities $P(\mu \to \nu)$ which define our Markov process. First we split them in a selection probability $g(\mu \to \nu)$ and an acceptance probability $p_{\mathrm{acc}}(\mu \to \nu)$. $g(\mu \to \nu)$ defines the probability to select the state ν as the next state in the Markov chain. Here we will always use symmetric selection such that the inverse pairs of selection probability are equal,[12] i.e., $g(\mu \to$

---

[12] Note that this is not necessary and the freedom to adjust the selection probabilities can lead to very efficient algorithms, e.g., the Wolff-cluster algorithm for ferromagnets [26]. However, in the work of this thesis uniform selection probabilities are used always and therefore selection probabilities will not be elaborated beyond this footnote.

$\nu) = g(\nu \to \mu)$. This leaves the acceptance probabilities, which need to fulfill

$$\frac{p_{\mathrm{acc}}(\mu \to \nu)}{p_{\mathrm{acc}}(\nu \to \mu)} = \exp\left(-\frac{E_\nu - E_\mu}{T}\right). \tag{2.19}$$

A very common choice fulfilling this relation, is the *Metropolis acceptance probability* [27]

$$p_{\mathrm{acc}}(\mu \to \nu) = \min\left\{\exp\left(-\frac{E_\nu - E_\mu}{T}\right), 1\right\}. \tag{2.20}$$

This formulation maximizes the acceptance probability and leads therefore to samples which change more quickly.

The generalization for arbitrary distributions p_μ instead of the Boltzmann distribution is called *Metropolis-Hastings algorithm* [28] and uses the generalized acceptance ratio (again under the prerequisite that the change move $\mu \to \nu$ is chosen with the same probability as $\nu \to \mu$)

$$p_{\mathrm{acc}}(\mu \to \nu) = \min\left\{\frac{p_\nu}{p_\mu}, 1\right\}. \tag{2.21}$$

We will encounter this acceptance probability again in Section 2.2.5.

The Markov process is started with an arbitrary state and iteratively new states are proposed, which are accepted according to Equation (2.20), such that the transition probability is correct. The new proposed changes should typically be chosen in a way that they on the one hand change the system sufficiently but are on the other hand often accepted. For the Boltzmann example, that means that their energies should typically be close to each other – especially at low temperatures. However, small changes cause consecutive configurations μ and ν to be correlated. Therefore, we have to run this Markov process for some time. Usually time for a MCMC simulation is measured in *Monte Carlo sweeps*, which is one change attempt per degree of freedom, since this is a rough estimate after how many changes two states could be reasonably decorrelated. Typically this decorrelation does take longer than one sweep. Especially in the beginning, when the starting configuration is far away from typical states, the Markov process needs to *equilibrate* for some time t_{eq}, before the samples follow the correct distribution.

Furthermore, this correlation has to be considered for the calculation of the error estimates of an observable S, otherwise one would drastically underestimate the uncertainty. To correct the error estimates for this fact, the integrated *autocorrelation time* [19]

$$\tau = \int_0^\infty \frac{\chi(t)}{\chi(0)}\,\mathrm{d}t \tag{2.22}$$

with the exponentially decaying *autocorrelation function*

$$\chi(t) = \int S(t')S(t'+t)\,\mathrm{d}t' - \langle S\rangle^2 \tag{2.23}$$

is calculated.[13] Since we only need an upper bound for τ, we can reduce noise by integrating only up to the first negative value instead of the whole time series. Then, only every 2τ-th measurement, which are statistically uncorrelated, are taken into account for the evaluation. From these data we can determine statistical errors for any quantity by standard methods for uncorrelated data, e.g., via bootstrap resampling [12–14].

Black Box Approach: Markov Chain of Random Number Vectors One of the most crucial ingredients of a Markov chain is the change move, which changes a configuration to another configuration with a slightly different energy E. This protocol is often very specific to the model. However, it is possible to use model-agnostic change moves, which do not operate on the realizations of the model, but on the random numbers used to create the realization in a computer. This is especially useful for growth models, for which it could be very hard to generate realizations uniformly by change moves. In this thesis, if a specialized change move operating on realizations of the model is used, it will be defined together with the model. Otherwise the following general method [29] is used and will throughout the thesis be referred to as the *black box approach*.

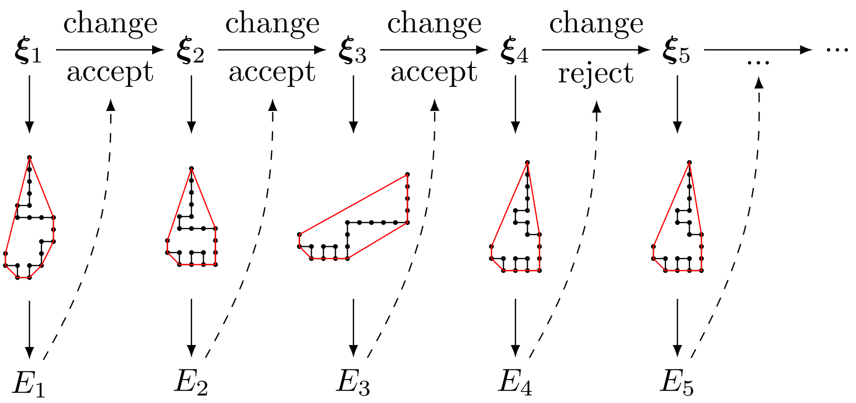


Figure 2.3.: Schematic of the black box Markov chain method. This method changes the underlying random numbers $\boldsymbol{\xi}_i$ and generates new configurations from scratch using the slightly modified random numbers. This is in contrast to the algorithms, which operate directly on the configuration.

To simulate any stochastic process on a computer, we need some source of random numbers, due to the deterministic nature of computers. While there are applications with a need for very high quality random numbers, like the generation of cryptographic keys, computer simulations only need numbers that appear random and whose correlations have no impact on the simulated problem – pseudo random numbers. Fortunately, there was much work invested to construct pseudo random number generators which are fast and yield random numbers of sufficient quality. All random numbers of

---

[13] Instead of the direct calculation in $\mathcal{O}(N^2)$, one can use a Fast Fourier Transform to calculate the autocorrelation function in $\mathcal{O}(N \ln N)$: $\chi(t) = \mathcal{F}^{-1}\left(|\mathcal{F}(S(t') - \langle S \rangle)|^2\right)$.

this work are, for example, created using the *Mersenne Twister* [30], which is a well tested algorithm in the context of Monte Carlo simulations. In the remainder of this thesis pseudo random numbers will be referred to simply as random numbers.[14]

To create a realization of, e.g., a random walk on a square lattice, one creates T random integers $\xi_i \in \{1, 2, 3, 4\}$ corresponding to the four possible directions (north, east, south or west) in which the i-th step will be taken. Note that the vector of random numbers $\boldsymbol{\xi}$ is a representation of the random walk equivalent to any other representation, e.g., the tuple of visited points. While this correspondence may be not as simple for every model, it is clear that the random numbers used to construct a realization always define the realization. This is the fundamental idea of a general change move. Instead of changing the realization of the model directly, e.g., a *crankshaft move* (cf. Section 3.3.1 and Figure 3.5) for the random walk on a lattice, the vector of underlying random numbers $\boldsymbol{\xi}$ is changed at one entry. The acceptance or rejection of this change is then determined by an observable $E \equiv S$ acting as an energy derived from the realization as visualized in Figure 2.3.

This technique, of course, has drawbacks. In the example of the random walk, there are specialized moves, like the crankshaft move. Typically, these are local moves which are generally faster to compute, while the change of the t-th entry in $\boldsymbol{\xi}$ will change the positions of the visited sites for all times $\tau \geq t$. This might even lead to large changes in the observable used as an energy and therefore to high rejection rates. In practice, the changes in the observable are often small and enable a fast evolution of the Markov chain. On the other hand this method is general and can be applied to, e.g., growth models, for which change moves preserving the correct statistics might be hard to devise.

2.2.4. Metropolis Algorithm to Sample the Large Deviation Regime

The previous section already introduced Markov chains with the classical example to generate Boltzmann distributed configurations using the Metropolis algorithm. Here we will explore how this method can be applied to obtain the probability density function of an arbitrary observable S over a large part of its support. That means, we can get reliable statistics down to very small probabilities, often below $p < 10^{-100}$. This explanation is guided especially by Reference [31]. This method is not novel and has already been applied to various different subjects from scores of DNA or protein sequence alignments [32–34] and non-equilibrium thermodynamic processes [29] over properties of random graphs [31, 35–38], to properties of convex hulls of random walks [39, 40].

First, we need to introduce an energy, which we will identify with the observable of interest $E \equiv S$. As should be intuitive from statistical mechanics, lower temperatures correspond typically to lower energies of the system. In the same way an artificial "temperature" Θ biases the simulated realizations towards lower energies S. The lower

---

[14] Especially since it does not play a role if they are pseudo or real random numbers as all introduced concepts can be applied regardless.

the absolute value of the temperature, the lower the typical energy. An infinite temperature will lead to uniform, i.e., unbiased samples, since Equation (2.20) will always be 1 and thus every change will be accepted. This, except for the autocorrelation, is basically simple sampling. To generate samples of higher energy we can formally introduce negative temperatures. For those Equation (2.20) will always accept increases and sometimes reject decreases in energy and therefore bias towards higher energies. In the top left panel of Figure 2.4 the time series for different temperatures are visualized and it is well visible that different temperatures typically fluctuate around different values of the energy S.

Now that we have a qualitative understanding of the bias, we have to analyze it quantitatively to derive the correct reweighting allowing us to obtain the actual unbiased distribution. At this point we will change the notation slightly to distinguish the probability $Q(c)$ that a configuration c is encountered in the unbiased ensemble from the probability $P(S)$ of a configuration with energy S in the unbiased ensemble. Applying the Metropolis algorithm introduced in the previous section, we arrive at the equilibrium distribution of the ensemble biased by the temperature, which is known to be

$$Q_\Theta(c) = Q(c)\frac{\mathrm{e}^{-S(c)/\Theta}}{Z_\Theta}, \tag{2.24}$$

where $Z_\Theta = \sum_c Q_\Theta(c)$ is the partition function of this artificial temperature ensemble. The connection between the desired distribution $P(S)$ and $Q(c)$ is easily obtained by summing the probabilities of all configurations which have the specified energy S'

$$P(S') = \sum_{\{c|S(c)=S'\}} Q(c), \tag{2.25}$$

where S' is a value and $S(\cdot)$ the function to obtain the energy of its argument. In the following (the same as before), the notation will be a bit more sloppy as we will refer to the value also with S. Analogously the probability to encounter a value S in the biased ensemble is

$$\begin{aligned} P_\Theta(S) &= \sum_{\{c|S(c)=S\}} Q_\Theta(c) && (2.26)\\ &= \frac{1}{Z_\Theta}\sum_{\{c|S(c)=S\}} Q(c)\mathrm{e}^{-S(c)/\Theta} && (2.27)\\ &= \frac{\mathrm{e}^{-S/\Theta}}{Z_\Theta}P(S) && (2.28)\end{aligned}$$

leading to

$$P(S) = \mathrm{e}^{S/\Theta} Z_\Theta P_\Theta(S). \tag{2.29}$$

Thus sampling the biased ensemble at Θ one obtains the histogram estimating $P_\Theta(S)$ which can be transformed into the unbiased distribution $P(S)$ up to the normalization

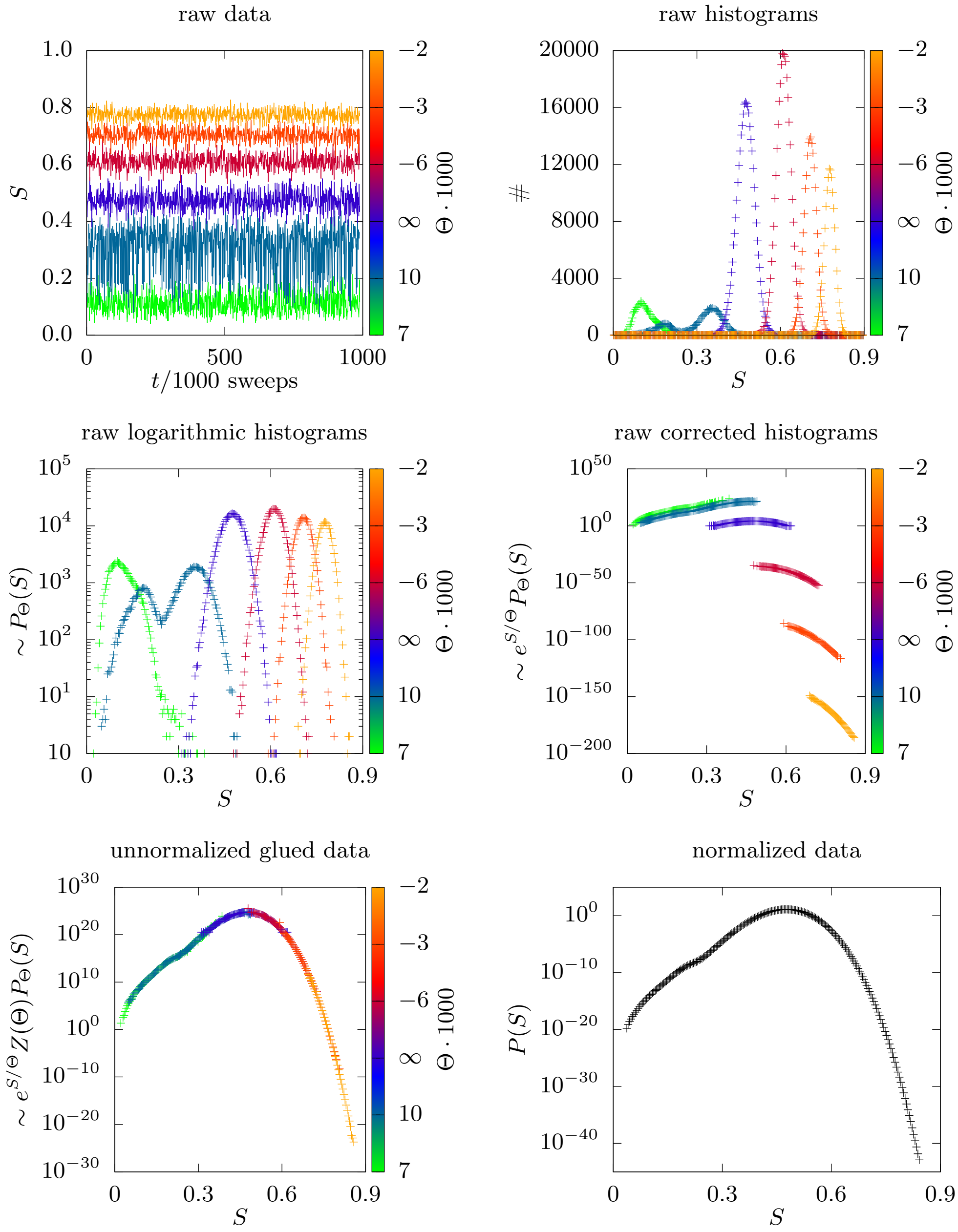


Figure 2.4.: Overview over the whole process of the temperature-based sampling scheme. Different temperatures Θ are visualized with different colors. Top left: time series of the Markov process. Note that the equilibration time is very short in this system and therefore not visible on the left side. Top right: simple histogram of the values encountered in the time series. Note that correlated measurements are already discarded. Middle left: the same histograms in logarithmic scale. Middle right and bottom left: intermediate steps of the correction step. Bottom right: normalized distribution over a large part of the support.

Z_Θ. Since the sampling is biased by Θ, the parameter Θ can be tuned to steer the samples into specific parts of the distribution, e.g., away from the main region of the distribution, such that good statistics can be obtained for these parts. This is well visible in the histograms of Figure 2.4.

To obtain Z_Θ, one can exploit the fact that the distribution is continuous. Therefore one needs to sample $P_\Theta(S)$ at multiple temperatures such that the distributions $P_{\Theta_i}(S)$ are overlapping. Given two overlapping distributions $P_{\Theta_{i-1}}$ and P_{Θ_i} the ratio of their free parameters $Z_{\Theta_{i-1}}/Z_{\Theta_i}$ is obtained by enforcing equality over the overlap

$$P_{\Theta_{i-1}}(S) Z_{\Theta_{i-1}} \mathrm{e}^{S/\Theta_{i-1}} = P_{\Theta_i}(S) Z_{\Theta_i} \mathrm{e}^{S/\Theta_i}, \tag{2.30}$$

In Figure 2.4 this can be seen as the shift of the curves in the fourth picture to result in the fifths. The technical details how this was achieved honoring the sampling errors is described in Appendix B.1.

This whole procedure is pictured in Figure 2.4, where all stages are visualized at an example of the largest *bi-edge-connected component* of an *Erdős-Rényi graph* with fixed number of edges M – a model that will be introduced in Chapter 6.

Since this method needs to simulate different temperatures anyway, a natural improvement is *parallel tempering*, which typically leads to shorter equilibration and autocorrelation times. It works by simulation of all temperatures Θ_i in parallel and exchanging system i and j with an acceptance probability of

$$p_\text{acc} = \min\left\{1, \exp\left[(S_j - S_i)\left(\frac{1}{\Theta_j} - \frac{1}{\Theta_i}\right)\right]\right\}. \tag{2.31}$$

This way, the configurations are simulated at different temperatures. At higher temperatures they can overcome energy barriers and cool down into a different valley when they are swapped back to a lower temperature. Typically this enables a better sampling of the energy landscape and produces therefore more reliable results. It is easy to test that this relation actually obeys detailed balance Equation (2.16) and therefore yields correct results [19]. However, in the studies belonging to this thesis parallel tempering was not used.

Error estimates of single bins can be obtained by bootstrap resampling [12–14]. Therefore the bootstrap samples are drawn from the initial time series data of the Markov process and the above evaluation is done for each of, say, 100 bootstrap samples, from which the single bins are evaluated to obtain error estimates. Generally, the errors turned out to be very small.

We use this biased sampling method for its simplicity and robustness, though for some problem instances the equilibration is difficult. For those cases Wang Landau sampling, introduced in Section 2.2.5, is used instead.

2.2.5. Wang Landau Sampling

The temperature-based sampling scheme above uses an artificial temperature to generate samples from the tails of the probability distribution. Though the temperatures need to be chosen carefully to generate samples covering the whole support.

If the distribution $P(S)$ was known in the beginning, one would know which configurations are improbable and should be preferred. A Markov chain accepting changes with

$$p_{\mathrm{acc}}(c_i \to c_j) = \min\left\{\frac{P(S(c_i))}{P(S(c_j))}, 1\right\} \tag{2.32}$$

would sample every S equally (cf. Equation (2.21)). A configuration with an improbable S would likely be accepted, while configurations with probable S are more likely to be rejected. A histogram $H(S)$ of the encountered S would be flat, leading to the name of this and similar methods as *flat histogram* methods.

Of course, $P(S)$ is not known beforehand, but is rather the result we want to obtain. However, given a good enough estimate $g(S)$ of the distribution, in this context also often called *density of states*, we will arrive at an almost flat histogram H, where every bin is visited often and the statistics is decent over the whole range. The deviations from perfect flatness can even be used to correct our estimate

$$P(S) = \frac{H(S)}{\langle H \rangle} g(S), \tag{2.33}$$

where $\langle H \rangle$ is the average number of entries of each bin. This is known as *entropic sampling* [41].

Wang Landau sampling (WL) [42, 43] builds on this technique and extends it by the ingenious idea to continuously improve the estimate of the (unnormalized) density of states $g(S)$ during the simulation. In the beginning $g(S)$ is chosen arbitrarily, for example as uniform $g(S) = 1$ and a *refinement factor* f is typically initialized as $f = \exp(1)$. In every iteration the current configuration c_i is changed to c' and accepted as $c_{i+1} = c'$ with

$$p_{\mathrm{acc}}(c_i \to c') = \min\left\{\frac{g(S(c_i))}{g(S(c'))}, 1\right\} \tag{2.34}$$

otherwise rejected, i.e., $c_{i+1} = c_i$. After every attempt the estimate g is updated as $g(S(c_{i+1})) \mapsto g(S(c_{i+1})) \cdot f$. Note that due to the multiplication by the constant $f > 1$ our estimate of the density of states g can approximate the actual distribution very quickly, even given very large relative differences of, say, a factor 10^{100}.

This is repeated until the auxiliary histogram $H(S)$ fulfills a *flatness criterion*, which is typically that the smallest bin has at least $0.8\,\langle H \rangle$ entries. As soon as this criterion is fulfilled, the refinement factor is reduced to $f \mapsto \sqrt{f}$ and the auxiliary histogram is reset. If the refinement factor f drops below f_{final}, the algorithm terminates and $g(S)$ is the estimate for the unnormalized probability density distribution. Consequently, the value chosen for f_{final} determines the quality of $g(S)$.

Note that the acceptance probability changes in every step and does therefore not obey detailed balance. This introduces systematic errors of the order of $\ln f_{\mathrm{final}}$. Though, there are methods to amend this behavior, which will be explained below.

Wang Landau sampling is most efficient when subdividing the support of S into *windows*. Each window i generates an estimate g_i for a part of the density of states. Monte Carlo moves which would leave their window are always rejected – though counted for the corresponding histogram bins [44]. This will sample the density of states in each window independently, which can be used to parallelize the computation. Further, smaller windows also reduce the time needed for the auxiliary histogram to become flat. Those fragments g_i need to be stitched together similar to the temperature-based sampling introduced in the last section. Therefore the windows need to overlap. The overlap can be used to discard the bins next to the border which are most susceptible to boundary effects. Bear in mind that this will be only correct, if there is a path of changes inside this window to generate every configuration of this window, i.e., ergodicity must be fulfilled for each window. Otherwise there would be unreachable configurations which may lead to subtly wrong results.

This leads quite naturally to a further improvement of the algorithm. Since the windows need to overlap anyway, two configuration in different windows can be proposed to be swapped, if both are in the overlapping region of the corresponding other window [45]. Such a swap of configuration μ with configuration ν from the windows i and j is accepted with

$$p_{\mathrm{acc}} = \min \left\{ \frac{g_i(S(\mu))}{g_i(S(\nu))} \frac{g_j(S(\nu))}{g_j(S(\mu))}, 1 \right\} \tag{2.35}$$

motivated by detailed balance. This *replica-exchange Wang Landau* typically leads to faster equilibration and shorter autocorrelation times and is conceptionally very similar to the aforementioned parallel tempering for the temperature-based approach. Though, all problems of this study were well behaved when treated with classical Wang Landau sampling such that replica-exchange Wang Landau was not used.

Since $g(S)$ will contain very small entries in the tails, such that common data types can not represent them, this algorithm should be implemented to operate on the logarithms of g and f. A pseudo code implementation could look like this.

```
def wang_landau(model, f_final=1e-8):
    H = Histogram()
    g = Histogram()

    f = 1
    while f > f_final:
        while not H.flatness_criterion():
            S_old = model.score()
            model.change()
            S_new = model.score()

            p_acc = exp(g[S_old] - g[S_new])

```

```
                if uniform_random_number(0., 1.) > p_acc:
                    model.undo()
                    S_new = S_old

                g[S_new] += f
                H[S_new] += 1
            H.reset()
            f /= 2;

        return g.normalized()
```

Technically, to obtain statistical error estimates, one repeats the algorithm a few times and takes a bin-wise standard error of multiple obtained distribution estimates.

In comparison to the temperature-based sampling scheme there are mainly two advantages. Sometimes there are barriers in the energy landscape, for example dividing coexisting phases in systems, corresponding to first order phase transitions in the artificial temperature ensemble [31]. It is usually hard for the temperature-based Metropolis changes to overcome them, such that they are stuck on one side for long periods of time, which leads to long autocorrelation times in the best case and wrong results in the worst. Also, the region between the two dominating phases are often hard to sample with the temperature-based approach, leaving gaps in the combined histograms, which makes it impossible to “glue” the distributions together. Since Wang Landau sampling operates directly on the energy landscape, it does not suffer from these problems and can easily sample those deep valleys.

Modified Wang Landau Sampling extended by Entropic Sampling

The two major downsides of Wang Landau sampling are its systematic error in the order of $\ln f_{\text{final}}$ and the flatness criterion, which makes it hard to estimate the runtime of the program. Both problems are addressed in References [46, 47], which introduce another schedule to change the value of f during the simulation. The first phase is quite similar to the original Wang Landau algorithm, but it performs a fixed number of sweeps and requires instead of a flatness criterion only that $H_i > 0 \ \forall i$ before f is reduced. The Monte Carlo time t is measured as the number of performed sweeps. As soon as $\ln f < 1/t$, the second phase starts and f is updated after every sweep to $\ln f = 1/t$. This means that the total number of sweeps is fixed by $t_{\text{total}} = 1/\ln f_{\text{final}}$. Reference [47] shows that this schedule does in fact eliminate the saturation of the error.

Since detailed balance is still not fulfilled, f_{final} needs to be quite small to yield good results, which often takes too much computing power in practice. For this study $t_{\text{total}} = 10^5$ could be obtained in a reasonable amount of time. But the estimate of g_{wl} obtained by this method can be used to perform the aforementioned entropic sampling [41, 48]. The estimate for the density of states g_{wl} is not altered during the entropic sampling, which grants detailed balance and eliminates all systematic errors. After a

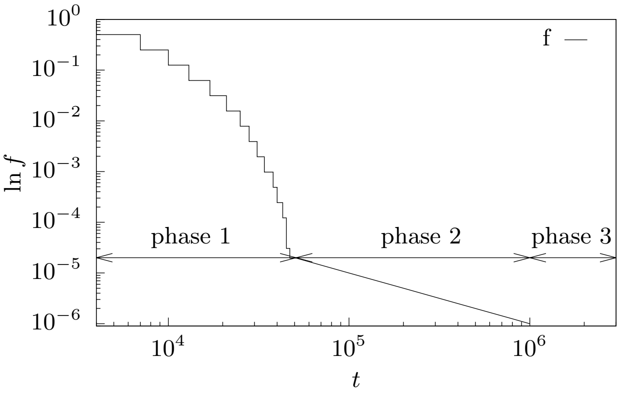


Figure 2.5.: Schedule of the used modified Wang Landau sampling. The first phase sees an exponentially decaying refinement factor f, the second phase is a power-law decay as suggested by Reference [46] and the third phase performs entropic sampling [41, 48], i.e., the estimate g is not changed anymore, corresponding to a constant refinement factor $f = 1$.

set amount of sweeps, here $2t_{\text{total}}$, the collected histogram H is used to get a better estimate of the density of states g using Equation (2.33), i.e.,

$$g(S) = \frac{H(S)}{\langle H \rangle} g_{\text{wl}}(S).$$

A typical schedule with $\ln f_{\text{final}} = 10^{-6}$ of the modified Wang Landau sampling is shown in Figure 2.5.

Comparison to the Temperature-Based Scheme

To demonstrate the quality of the introduced algorithms, we sample a random walk on a square lattice with $T = 256$ steps and handle the area A of its convex hull (cf. Section 3.3.2) as the energy. For this simple model the maximum area of the convex hull $A_{\max}$ is reached by an "L"-shaped configuration with $A_{\max} = \frac{1}{2}(T/2)^2 = 8192$. There are 8 configurations of maximum area due to the lattice symmetries and $4^{256} \approx 10^{154}$ total configurations, because at each of the 256 steps the walk can choose 1 of 4 directions, which we can use to calculate the probability of $P(A = 8192) \approx 10^{-153}$. Since the bins have some width, we can not reproduce this value exactly, but we can expect $\min_A \{P(A)\} \gtrsim 10^{-153}$. In fact, a comparison in Fig. 2.6 of the temperature-based approach and the two Wang Landau sampling variants shows that all distributions match both the expectations and each other rather nicely. The unmodified Wang Landau algorithm shows no strong deviations in this test, but uses far more time in the rightmost window than the improved Wang Landau sampling.

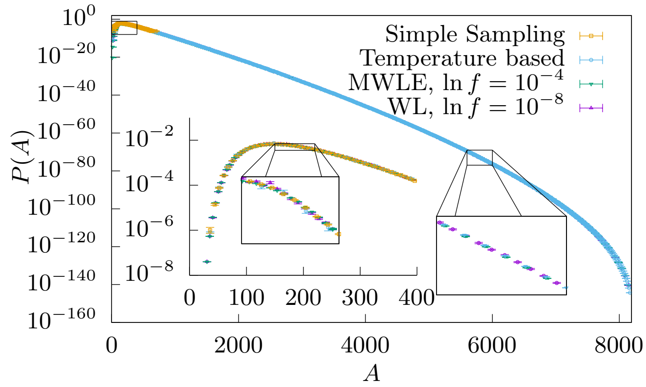


Figure 2.6.: Comparison of the unmodified Wang Landau sampling, the modified version used here (MWLE) and a temperature-based approach, for the volume of the convex hull of a $T = 256$ random walk on a square lattice. Note that the steep decline on the right side is caused by the lattice structure, where simply no larger realizations exist. The insets zoom closer to show that all methods yield within errorbars compatible results. For clarity not every data point is plotted. The simulation for these test cases finished in a few hours CPU time.

3. Convex Hulls of Random Walks

This chapter is about the properties of *convex hulls* enclosing the trajectory of *random walks*. The convex hull $\mathcal{C}$ of a set of points $\mathcal{P}$ is the smallest convex polytope including all points of the set $\mathcal{P}$, which consists in this case of all points the random walk visited. This is a field of mostly mathematical interest but with applications in ecology, where the random walk is interpreted as the trajectory of an animal[1] and the convex hull is an estimate for its home range [53–55].

Since the random walk is arguably one of the most simple models of stochastic geometry and convex hulls are one of the most simple geometrical objects suited to characterize point sets, their combination lends itself nicely to mathematical scrutiny, though it is far from trivial. The convex hull of a stochastic process connects interestingly extreme-value statistics with geometry, since a convex hull consists of – in some sense – the extreme points of the point set (cf. Section 3.1.1).

We will start this chapter with a non-exhaustive review of previous work studying the convex hulls of random walks in Section 3.1. We will especially focus on one publication, which introduced an analytical approach simple enough to be explained on a few pages of this thesis in Section 3.1.1. Using this background, we will proceed to formulate our research question in Section 3.2. To study this question, we will introduce all random walk models under scrutiny in Section 3.3.1 and computational tools to construct convex hulls in Section 3.3.2. Finally Section 3.4 gives a short overview over the results published in Articles A.1 to A.3 about this topic.

3.1. Current State of Research

The two properties of the convex hulls of random walks which attracted most interest in the past and also in this study, are the perimeter and the area of convex hulls or their higher dimensional analoga. While the mean perimeter for planar random walks in the limit of large walk lengths is known since about 40 years [56], the mean area is only known since 25 years [57]. 10 years ago a simpler method to calculate both area, perimeter and more observables, which is also applicable, e.g., to the joint convex hull of multiple independent walks and can, in principle, be extended to higher dimensions, was published [58, 59]. This method is based on Cauchy's formula, which relates the support of a closed curve to the perimeter and area enclosed by the curve. A quick

---

[1] In fact, the first mention of a *random walk* by this name was in 1905 in a letter to Nature by Karl Pearson [49–51], who wanted to use it as a model for mosquitos. Interestingly, the solution to his question was already found 25 years prior by John William Strutt (also known as Lord Rayleigh). However, the phenomenon of Brownian motion was of course known far before. The first rigorous study of random walks on lattices is to be attributed to Georg Pólya [52].

overview over the idea of this approach will be shown in Section 3.1.1. Even more recently a general closed formula for the mean hypervolume and surface in arbitrary dimensions was found [60]. As was an exact combinatorial formula for the mean volume of finite length walks with Gaussian distributed step lengths [61]. Despite this consistent progress regarding the mean values, higher moments seem to be out of reach – in fact, there is only one exact result known for the variance of *Brownian bridges* [62], which are random walks with the further constraint that the last step needs to end on the initial starting point, i.e., the walk needs to be closed. Further, only bounds on the variance are known, e.g., for the perimeter of walks with identically, independently distributed jumps of finite variance $\sigma_X^2 + \sigma_Y^2$ we have $\mathrm{Var}(L_N) \leq \frac{\pi}{2} N \left(\sigma_X^2 + \sigma_Y^2\right)$ [63].

It should also not be unmentioned that the perimeter and area of the convex hull are not the only quantities of interest. For example the number of facets (see Section 3.3.2 for a precise definition) of the hull or the number of points which constitute the hull were of interest since almost 60 years [64].

For a more throughout review of the approaches used, especially by some of the earliest publications mentioned here, Reference [59] is highly recommended.

Besides the pure mathematical interest, as mentioned, there are also applications to animal movement, e.g., as a model for the spatial extend of animal epidemics. In Reference [65] a disease in the spirit of the *SIR model* [66] is modeled. SIR is a simple mean field like model of coupled differential equations, where a portion of the population is either susceptible to, infected by or recovered from (and then immune to) a disease. Often it is also studied (discretized) on complex networks [67, 68], where the network topology determines which individuals can infect each other.

Here, a variant is modeled using a two dimensional branching Brownian motion[2] similar to the one shown in Figure 3.1, where a Brownian motion can branch into two Brownian motions or perish with some probability. This model assumes a uniform distribution of individuals on the plane. The branching represents the infection of a susceptible individual by another infected individual with probability b per timestep (or $b\,\mathrm{d}t$ for the continuous case) and the perishing represents the recovery with probability γ.

The convex hull of this branching Brownian motion is used as an estimate for the area over which the infection has spread. This is not a sensible model for human behavior, since the movement patterns of humans can not be modeled by Brownian motion well, but are dominated by small-world effects, e.g., air travel allows non-local spreading of the disease (see also Chapter 6). Nevertheless for animal or even plant diseases, where the infection is mediated by insects, this is reasonable.

The classical SIR model has a parameter $R_0 = \beta N/\gamma$, where β is the infection rate and γ the recovery rate, which indicates its behavior. In the thermodynamic limit

---

[2] Unfortunately, the word “Brownian motion” is used for multiple processes in the literature and can mean either the original movement of small particles caused by molecular dynamics, a time continuous stochastic process also known as *Wiener process* or a time discrete random walk with a Gaussian jump length distribution, also called *Gaussian random walk*. Since the realizations of the last two are statistically identical, this should not cause too much confusion in this chapter, despite both processes being referred to with the same name.

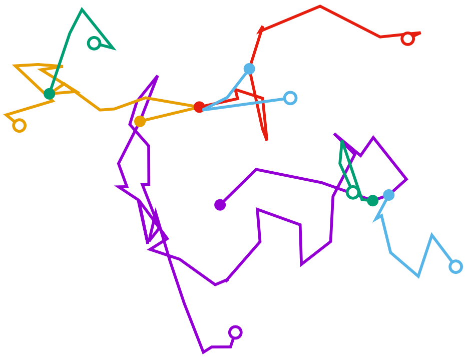


Figure 3.1.: Example of a branching Gaussian walk, a time discrete version of the model used in Reference [65]. Different individuals are visualized with different colors. At the point of infection the walk branches, marked with a dot. When the individual recovers, the corresponding branch ends, marked with a circle. (Some colors are used twice.)

of $N \to \infty$ individuals, the disease will vanish before infecting $\mathcal{O}(N)$ of the system for $R_0 < 1$, which is called the subcritical phase. For the supercritical phase $R_0 > 1$ the disease will infect a finite fraction of the population. Dumonteil *et al.* showed in Reference [65] that the same behavior is observable for the area and perimeter of the convex hull around the branching Brownian motion with the same critical point $R_0 = b/\gamma = 1$. Moreover, they determine the mean perimeter and area, using the approach based on Cauchy's formula (cf. Section 3.1.1), in the supercritical regime. They show that the convex hull grows ballistically, i.e., the perimeter grows linear in time and the area quadratically in time, which means that the region where the disease is spread grows far faster than the underlying diffusive process might have suggested.

Another recent application of convex hulls around random walks is to differentiate phases of intermittent stochastic processes [69]. In nature there are many movement processes, which change their phase in an intermittent way, e.g., bacteria show "run and tumble" [70][3] phases, i.e., "run" phases in which the change of direction is smooth and "tumble" phases in which the change of direction is abrupt. Note that Reference [69] cites more occurrences of this intermittent behavior, like animal foraging, which changes between fast movement and slow searches, and many occurrences in microbiology. It is important to differentiate between the phases of intermittent processes, to not reach wrong conclusions, e.g., interpreting an intermittent change of diffusive and ballistic phases in the trajectory of a particle as anomalous diffusion. Reference [69] proposes a method utilizing the *local convex hulls* to discriminate different phases. "Local" means here that of the time series of positions $\boldsymbol{x}(t)$, for every point its τ predecessors and successors are used to calculate their joint hull around $2\tau+1$ points. An example where two local convex hulls are visualized for a small part of an intermittent process is shown in Figure 3.2. Of these hulls the volume and diameter, i.e., the largest distance between any pair of points, are calculated. This transforms a d-dimensional

---

[3] In the cited paper called "run and twiddle".

time series into two 1-dimensional time series. Of course, τ must be chosen carefully. A too small value of τ will result in greater noise, whereas a too large value might be longer than a phase lasts and therefore obscure it. To distinguish two phases, threshold values of the two observables are proposed. A comprehensive validation on different types of simulated intermittent processes shows the viability of this approach.

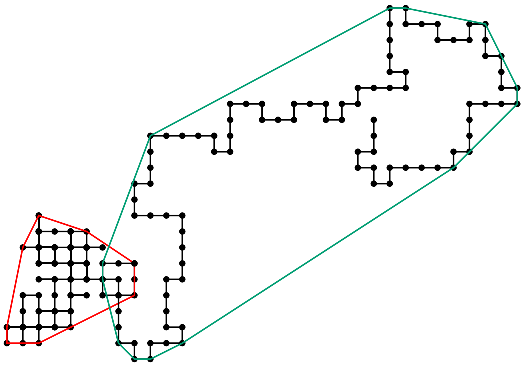


Figure 3.2.: Example of intermittent behavior detectable by local convex hulls. This example is a realization of a diffusive random walk which switches after $T_1 = 100$ steps to the anomalous diffusive loop-erased random walk (cf. Section 3.3.1) for $T_2 = 100$ steps. The two local convex hulls are constructed at times $t = 50$ and $t = 150$ with $\tau = 25$ and visualize nicely very different behavior, which can be used to differentiate the two phases.

While most of the above introduced work is of analytical nature, there are also a few prior publications studying the convex hulls of random walks by simulations. Especially noteworthy are two publications addressing the lack of results for higher moments by numerically obtaining large parts of the distribution of standard random walks in the plane [39] and multiple standard random walks [40]. These publications are important to the thesis at hand since they not only use similar methods, but also have similar aims as some publications constituting this thesis.[4] Both publications focus their work on large-deviation properties of the distributions of area and perimeter of convex hulls of random walks. This way not only the behavior of multiple random walks [58] was confirmed, but also numerically a large deviation principle (cf. Section 2.1) was established. Prompted by this study, Reference [71] proved the existence of a large deviation principle for a class of jump distributions rigorously.

3.1.1. Analytic Approach with Cauchy's Formula

Here we will explore the fundamental ideas of the approach followed in References [58, 59] to obtain analytically the mean perimeter and area of planar convex hulls of random point sets. This chapter will allow us to look into some of the connections of this problem to seemingly unrelated branches of mathematics and therefore establish

---

[4] Both of these References [39, 40] were published by members of the same working group I am part of and Reference [40] was even funded by the same project funding me.

the background of this problem. This chapter will borrow the notation of the review article Reference [59]. The first author of Reference [59], Satya N. Majumdar, is also coauthor of three of the publications belonging to this thesis.[5]

The central piece of this calculation is *Cauchy's formula* [72] named after Augustin-Louis Cauchy[6], also called *Cauchy-Crofton* formula named after Morgan Crofton, who applied the originally purely geometric formula to stochastic problems. Cauchy's formula connects the *support function* $M(\theta)$, i.e., a one dimensional projection, of a closed convex curve $\mathcal{C}$ to its perimeter L and area A via

$$L = \int_0^{2\pi} M(\theta)\,\mathrm{d}\theta \tag{3.1}$$

$$A = \frac{1}{2}\int_0^{2\pi} \left[M^2(\theta) - \left(M'(\theta)\right)^2\right]\,\mathrm{d}\theta, \tag{3.2}$$

where M' is the first derivative. The value of $M(\theta)$ at some angle θ is the extreme value of a projection of $\mathcal{C}$ to a line with angle θ through the origin, i.e.,

$$M(\theta) = \max_{(x,y)\in\mathcal{C}} \left\{x\cos(\theta) + y\sin(\theta)\right\}. \tag{3.3}$$

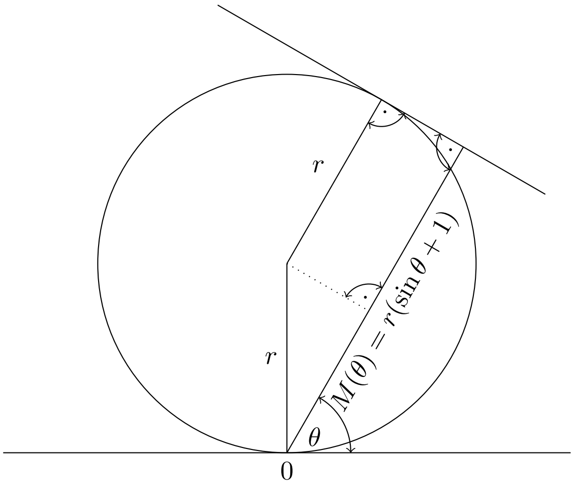


Figure 3.3.: The support function M of a circle with radius r standing on the coordinate origin.

We will start with a simple example from Reference [59] to test this with a closed convex curve, whose perimeter and area we know. Figure 3.3 shows a circle and its

---

[5] I had the opportunity to stay for the September 2016 in Orsay near Paris at the same institute, the Laboratoire de Physique Théorique et Modèles Statistiques (LPTMS).

[6] Augustin-Louis Cauchy was a very productive mathematician such that the term “Cauchy's formula” does not name a unique theorem – in fact the English Wikipedia has a “List of things named after Augustin-Louis Cauchy” with 48 entries at the time of writing (20.9.2018).

support function $M(\theta)$. If we put this into Equations (3.1) and (3.2), we arrive at

$$\begin{aligned} L &= \int_0^{2\pi} r\left(1+\sin\theta\right)\,\mathrm{d}\theta = 2\pi r \\ A &= \frac{1}{2}\int_0^{2\pi} r^2\left(1+\sin\theta\right)^2 - r^2\cos^2\theta\,\mathrm{d}\theta = \pi r^2 \end{aligned}$$

which are the values we would expect for a circle. A nice sketch of a proof, requiring only basic geometric and trigonometric understanding, for polygonal curves (which can be generalized to edges of infinitesimal length, i.e., continuous curves) is given in the appendix of Reference [59].

Since a convex hull is a smooth polygon, this formula could already be used for single instances of convex hulls. Note that the support function $M(\theta)$ of a convex hull of a point set $\mathcal{P}$ can be simply written as

$$M(\theta) = \max_{(x_i,y_i)\in\mathcal{P}} \left\{x_i\cos(\theta) + y_i\sin(\theta)\right\}. \tag{3.4}$$

The maximum projections will always be the projections of the vertices which are part of the convex hull. However, we are not interested in the perimeter of single hulls, but in the mean perimeter $\langle L\rangle$ over an ensemble of random realizations. Apparently, the x_i and y_i will become random variables in this case and consequently the support $M(\theta)$, too. The average over all realizations of the ensemble $\langle\cdot\rangle$ is a linear operation, hence we arrive at

$$\langle L\rangle = \int_0^{2\pi} \langle M(\theta)\rangle\,\mathrm{d}\theta \tag{3.5}$$

$$\langle A\rangle = \frac{1}{2}\int_0^{2\pi} \left[\left\langle M^2(\theta)\right\rangle - \left\langle \left(M'(\theta)\right)^2\right\rangle\right]\,\mathrm{d}\theta. \tag{3.6}$$

The maximum operation over a random variable in Equation (3.4) is then a sure sign of a relation to *extreme-value theory*. Using extreme-value theory one can derive a lot of interesting properties about random convex hulls for whole classes of random point sets. Here, we will not dive deeper into this topic. A more in depth look can be found in Reference [59].

For the case of Brownian motion the x and y components of the steps are identical and independently Gaussian distributed. Since the walk with this jump distribution is isotropic, considering the point $(x(\tau),\ y(\tau))$ at time τ, we see that the projection $z(\tau) = x(\tau)\cos(\theta) + y(\tau)\sin(\theta)$ is also a one dimensional Brownian motion itself. Especially, due to the isotropy, its support is statistically independent of θ, i.e., $\langle M(\theta)\rangle = \langle M(0)\rangle = \langle \max_\tau\{z(\tau)\}\rangle$. Therefore Equation (3.5) simplifies to

$$\langle L\rangle = 2\pi\langle M(0)\rangle. \tag{3.7}$$

The distribution function of the maximum of Brownian motion with T steps is known explicitly as $P(x\le M) = \mathrm{erf}\left(\frac{M}{\sqrt{2T}}\right)$ from which we can calculate the mean $\langle M(0)\rangle =$

$\sqrt{\frac{2T}{\pi}}$ and in turn with Equation (3.5) $\langle L \rangle = \sqrt{8\pi T}$. A slightly more involved, but in principle the same, procedure can be used to obtain $\langle A \rangle$.

This method is applicable to any random point sets whose distribution of the support function, or at least the first moment, is known (and the second to obtain the area). In Reference [58] it was therefore applied to the joint hull of multiple walkers. However for many random walk models, e.g., self avoiding-walk models, which avoid their past trajectory, not even the first moment of the support function $M(\theta)$ is known exactly, motivating the numerical studies Articles A.2 and A.3.

3.2. Research Question

The previous numerical work leaves open two obvious questions. The numerical study Reference [39] proposes a scaling of the whole distribution of perimeter L and area A in the same way as their respective means, i.e., $L \propto T^{\nu}$ and $A \propto T^{2\nu}$, and shows numerical evidence in two dimensions. The same article also gives an argument for the behavior of the rate function (cf. Section 2.1) in two dimension. This leads to the research question:

> Can the predictions about the scaling and rate function of the distribution of the properties of the convex hull of two dimensional standard random walks be generalized to standard random walks in higher dimensional spaces?

This question is the fundamental motivation of the research project which resulted in the publication of Article A.1.

Considering the wealth of literature about more complicated random walk models and the lack of results covering their convex hulls, the second question follows naturally:

> Can the predictions about the scaling and rate function of the distribution and the mean values and variances of the area and perimeter of the convex hull of standard random walks be generalized to other types of random walks? What are the mean values and variances of the area and perimeter for the convex hulls of self avoiding walks?

The answers to these questions are far less clear than the answer to the first question. While standard random walks do not change their behavior at higher dimensions, the behavior changes drastically for random walks which interact with their past trajectory. Since there are many different types of random walks, the study of a selection of them seems worthwhile. This motivated the research projects which resulted in Articles A.2 and A.3.

3.3. Models and Methods

This chapter will introduce the methods used in Articles A.1 to A.3 in a far more throughout manner than it is possible in an article. Therefore some fundamental concepts will be introduced and a lot of examples will be given. Some are not directly applied in the published papers, but give an overview over the topic and are included for this reason. While we already showed one approach to handle them mathematically in Section 3.1.1, we will focus on the handling of random walks and convex hulls within a computer in Sections 3.3.1 and 3.3.2.

3.3.1. Random Walks

One possibility to define a time-discrete random walk is using its T step vectors $\boldsymbol{\delta}_i$ of dimension d such that the position of the random walk at time τ is given as

$$\boldsymbol{x}(\tau) = \boldsymbol{x}_0 + \sum_{i=1}^{\tau} \boldsymbol{\delta}_i, \tag{3.8}$$

where $\boldsymbol{x}_0$ is the starting position, e.g., the coordinate origin. With this notation a random walk is uniquely defined by a tuple of steps $(\boldsymbol{\delta}_1, \dots, \boldsymbol{\delta}_T)$ and the points it visited are the elements of the set $\mathcal{P} = \{\boldsymbol{x}(0), \dots, \boldsymbol{x}(T)\}$.

Depending on the type of the random walk different constraints apply to the steps $\boldsymbol{\delta}_i$ and the resulting walk will have different properties. As mentioned in Section 2.2 we need to be able to introduce small changes into a given realization of a walk to enable Markov chain Monte Carlo sampling. The *naive method* is to replace one of the step vectors $\boldsymbol{\delta}_i$ with a different step vector $\boldsymbol{\delta}'$, which is equivalent to the black box approach (cf. Section 2.2.3). Indeed, this naive method turned out to be sufficient for some studied types of random walk. We will introduce specialized change moves if they are better suited for specific walk types in their respective sections.

In the following paragraphs all types of random walks relevant for this study will be defined, visualized and some particularities discussed. An important property is that most simple random walk models are characterized by a single exponent ν, which describes the growth of the end-to-end distance R as a function of the number of steps taken T, i.e., $R \propto T^\nu$. Its inverse is the fractal dimension $d_f = 1/\nu$ of the walk. Intuitively, it governs how fast the random walk leaves its starting point. After the introduction of the different random walk models of interest for this study, we will look into the exponent ν in more detail and compare it between all introduced random walk models.

Lattice Random Walk

The *lattice random walk* (LRW) is a walk on a square lattice (or hypercubic lattice in higher dimensions d) with step length 1. This is usually just called 'random walk,' but here a different notation is chosen for clarity. At each time step the walk proceeds to one of its $2d$ neighbors. An example is pictured in Figure 3.4 and a partial decision tree visualizing the probabilities to construct a specific configuration is shown in Figure 3.6(a). This walk converges to Brownian motion in the large T limit.

For the simple lattice random walk, there are more sophisticated move sets than our naive approach mentioned above that allow a MCMC simulation, especially local

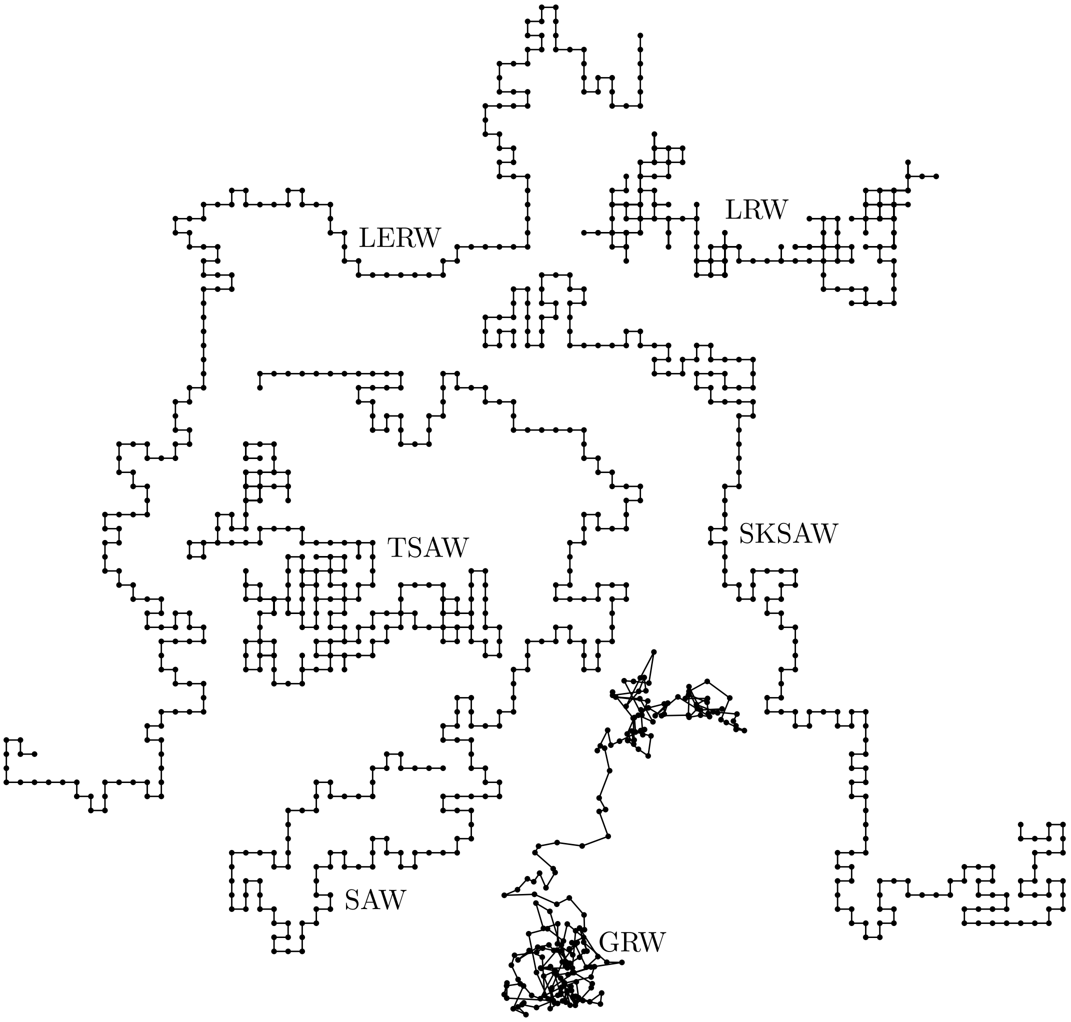


Figure 3.4.: Examples for different types of random walks with each $T = 200$ steps in $d = 2$. Pictured are lattice random walk (LRW), Gaussian random walk (GRW), self-avoiding random walk (SAW), loop-erased random walk (LERW), smart-kinetic self-avoiding walk (SKSAW) and "true" self-avoiding walk (TSAW), which will be explained in the following sections.

moves which only change a few point positions $\boldsymbol{x}(\tau)$, such that they usually can be performed faster than the naive method from above. Two examples of local moves are pictured in Figure 3.5. However, since for this study we are interested in the convex hull, which is a global property, we have to consider every single point after every change move such that local changes have only very minor benefit in this case. So for the simulations performed in context of this thesis, the naive changes of the black box approach turned out to be sufficient and not to slow down the simulations noticeably.

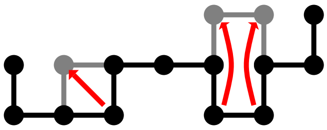


Figure 3.5.: Examples for two different change moves. The classical local move, which changes only one entry of the set of visited points $\mathcal{P}$ (left) and the crankshaft move, which always changes two entries of the set of visited points $\mathcal{P}$ (right).

Gaussian Random Walk

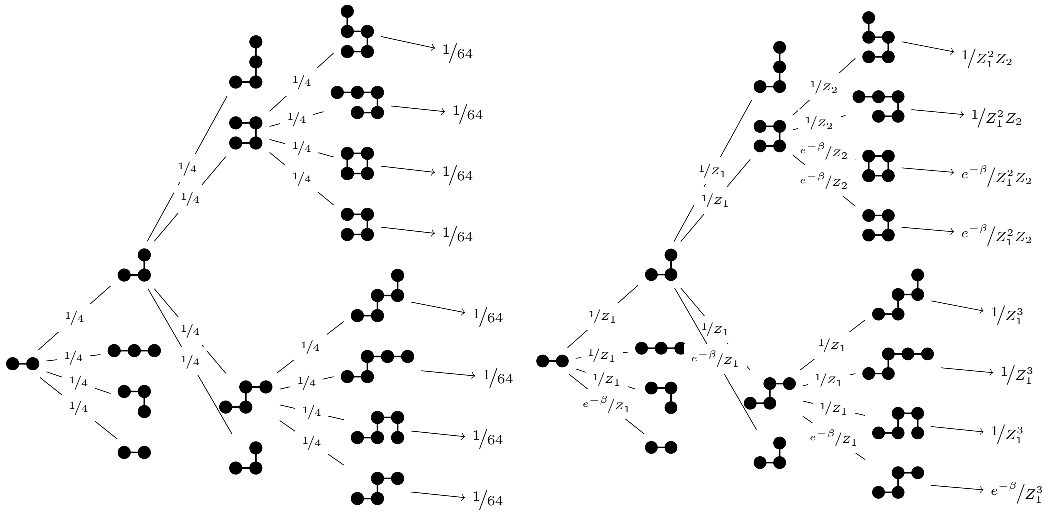
The *Gaussian random walk* (GRW), also often just called 'random walk,' lives in a continuous space. At each time step its next position is determined by a displacement vector drawn from an uncorrelated multivariate normal distribution with zero mean and width of $\sigma = 1$. An example is pictured in Figure 3.4. Note that this construction, despite being time discrete, leads to the same distribution of realizations as the time continuous Brownian motion. This walk is the only one studied in this thesis which does not live on a lattice. While the lattice random walk converges also to a Brownian motion for large values of T, the lattice structure usually still has influence at finite system sizes [39], called *finite-size effects*. On the other hand, the lattice structure simplifies the self-interacting walks introduced in the following sections greatly.

Self-Avoiding Walk

The *self-avoiding random walk* (SAW) is like the lattice random walk, but it may not visit any lattice site twice. This ensemble is defined by the uniform occurrence of all configurations which fulfill this criterion. For clarity, a complete enumeration of all configurations of length $T = 5$ (without rotational symmetric configurations) is shown in Figure 3.6(d) and an example for $T = 200$ is pictured in Figure 3.4.

This poses some difficulties when sampling self-avoiding random walk configurations. The naive and *wrong* approach would be to draw one random step after the other and in the case that this step would step onto the past trajectory, discard the last step and try a different one. This method has the problem that it quickly leads into *traps*, where it is not possible to continue the walk, since all neighboring sites were already visited. But more importantly, even without the traps, it does not generate the desired

(a) RW (b) TSAW



(c) SKSAW (d) SAW

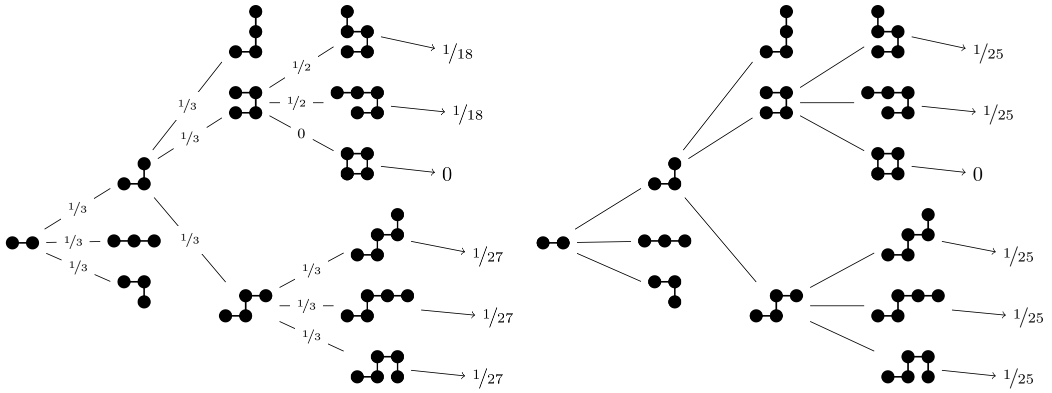


Figure 3.6.: An incomplete tree of decisions to generate a $T = 5$ random walk of the specific type. Rotational symmetric configurations are ignored.

distribution of walks. To sample uniformly from the ensemble of self-avoiding random walks, each configuration should be drawn with the same probability. The naive method prefers some configurations above others. In Figure 3.6(c) a decision tree of the naive method is shown, to clarify this problem. There it is clearly visible that the closely winded configurations on the top half of the image have a higher probability to be generated than the configurations shown on the bottom half.

To sample uniformly from all self-avoiding random walk configurations, the simplest correct method would be to take one random step at a time and discard *all* steps so far, if the next step would visit an already visited site. Since there is a chance at every step to visit a forbidden site, the chance to only step on free sites decays exponentially in the steps taken T, which leads to an exponential running time of this algorithm to generate a self-avoiding random walk of length T.

This can be mitigated by using Markov chain Monte Carlo based sampling methods. Starting from an initial realization, a typical Markov chain Monte Carlo simulation can be performed. In this study two kinds of trial moves are performed, namely the same naive changes mentioned earlier and *pivot moves* [24, 73]. The latter are performing a symmetry operation of the lattice (rotations, mirroring) around one randomly chosen pivot site as visualized in Figure 3.7. On the one hand, these changes are quite large and will often result in overlaps in which case the change is rejected. On the other hand, if it is accepted, the large change decorrelates subsequent samples quite fast. There are estimates for the ratio f of pivot moves which result in a valid walk configuration to scale as $f \propto T^{-p}$ with the number of steps T and $p \approx 0.19$ [73]. This typically results in a polynomial scaling in T of pivot move attempts needed to arrive at a decorrelated state. This Markov chain Monte Carlo approach can be directly combined with both large deviation sampling techniques introduced in Section 2.2. Therefore, change moves which do not lead to intersections and would be accepted by the pivot algorithm are rejected with probability $1 - p_{\text{acc}}$.

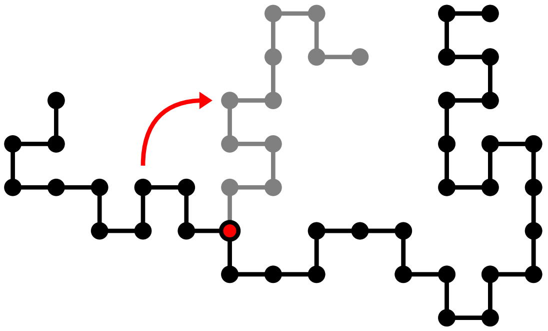


Figure 3.7.: Example of one pivot change. The used symmetry operation is $-\pi/2$ rotation around the red marked pivot site.

To create the initial realization the *dimerization* [24, 74] method can be used, which still has an exponential worst-case time-complexity but is in practice much faster than the naive approach. Dimerization is a divide an conquer approach, where two short self-avoiding random walks are combined to a longer self-avoiding random walk. If an intersection is detected, everything is discarded and the construction is started anew. The second option is to start from a straight walk, which is surely not self intersecting and equilibrate it using the above mentioned change moves. For our temperature based large deviation sampling scheme, depending on the walk length T and artificial temperature Θ the efficiency of these methods varies. In Article A.2 we started with a realization created by dimerization.

For this study we chose a rather simple implementation of the pivot algorithm. The state of the art method to perform pivot moves is based on an ingenious datastructure called *SAW-tree* [75, 76]. This allows single MCMC change moves to be performed in time $\mathcal{O}(\log N)$. In the context of this work, the time complexity of our MCMC change moves is dominated by the calculation of the "energy" in $\mathcal{O}(N \log N)$ (cf. Section 3.3.2). Thus, a simple implementation using a *hash set* suffices. After choosing the pivot, all sites belonging to the shorter end are erased from the hash set, transformed, tested against the remaining entries of the hash set and added to the hash set if no intersection arises. All these operations, and undoing them in case of rejection, have an amortized time complexity of $\mathcal{O}(N)$ and should therefore not bottleneck the simulations.

In higher dimensions the self-avoiding random walk has more directions in which it can go and thus more space to explore. Since its trace is more "diluted" it does not interact with itself as strong. Indeed for $d > 4$ the behavior is statistically the same as the lattice random walk, in the sense that the exponents ν are the same. Since for even higher dimensions the exponent does not change anymore, $d = 4$ is called the *upper critical dimension* of the self-avoiding random walk.

Loop-Erased Random Walk

The *loop-erased random walk* (LERW) was invented as a simple version of the self-avoiding random walk, in the hope that it would show the same statistical properties as the self-avoiding random walk [77]. This was mainly prompted since at that time the simulation of self-avoiding random walks was still not efficiently possible, e.g., neither PERM nor the pivot algorithm were invented. This hope, however, was not fulfilled, as the loop-erased random walk shows a different exponent ν and is therefore fundamentally different from the self-avoiding random walk. Nevertheless, it found some applications for some other seemingly unrelated problems [78], e.g., the uniform generation of spanning trees[7] can be mapped onto a loop-erased random walk on the graph. This correspondence was used to transfer known properties from one object to the other [79].

The loop-erased random walk is a simple lattice random walk, but when it crosses an already visited site i all steps since the last visit of i are discarded, i.e., the loop is erased. This is shown for a small example in Figure 3.8 where the gray nodes of

---

[7] A spanning tree of a graph is a subgraph without loops, which contains every node of the original graph.

a lattice random walk are erased to form a loop-erased random walk. After the loop erasure, the walk does not show any crossings. An example is pictured in Figure 3.4.

Finding a change move for this type of walk is not trivial. The naive change of the lattice random walk might lead to new loops and can therefore lead to shorter walks. Instead, we use the black box approach (cf. Section 2.2.3). The random number vectors $\boldsymbol{\xi}_i$ have no fixed length and new random numbers are generated if new loops are erased. Since there are also changes leading to a loop not being erased, not all random numbers are used for every realization. To maintain high acceptance rates, the changes should not introduce too much difference between consecutive realizations. Therefore it is necessary that we maintain the unused random numbers at the end of $\boldsymbol{\xi}_i$, i.e., $\boldsymbol{\xi}_i$ only grows and never shrinks. This should reduce the amount of change happening at one change move. Note that the loop erasure dynamic can lead to large changes in the realizations. For example, imagine a change would lead the T-th step to visit the origin again. The loop erasure will erase every single step and the construction basically starts from scratch. Despite this apparent difficulty, the black box approach works remarkably well in practice.

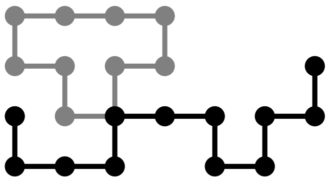


Figure 3.8.: Example of a loop erasure. The erased loop is shown in gray and the remaining loop-erased random walk in black.

To detect crossings in constant time, we use a hash set of all currently occupied sites. Maintaining it, requires to delete $\mathcal{O}(N)$ entries at each loop erasure, such that the time complexity of this change move is similar to the self-avoiding random walk $\mathcal{O}(N)$.

Analogous to the self-avoiding random walk, loop erasure becomes rarer in higher dimensions. Above the upper critical dimension $d = 4$ it becomes equivalent to the lattice random walk.

Smart Kinetic Self-Avoiding Random Walk

The *smart-kinetic self-avoiding random walk* (SKSAW) [80, 81] is a growing random walk which does not enter occupied sites and avoids getting trapped in loops. This also means that it is similar to the naive and wrong method without rejection first mentioned for the self-avoiding random walk above. The resulting probabilities of the configurations differ therefore from the self-avoiding random walk. First, configurations in which the walk is trapped, which are allowed for the self-avoiding random walk as long as no intersection occurs, do not occur in the smart-kinetic self-avoiding random walk, leading to relatively more configurations with large holes in them. While this effect might suggest that smart-kinetic self-avoiding random walk are on average more spread out, a second effect dominates this behavior. This stronger effect can best be understood when looking at the (partial) decision tree in Figure 3.6(c). In this tree it is visible that the distribution of the configurations is not uniform and closely winded configurations are more probable than for the self-avoiding random walk. This leads to a more compact form, which is also reflected in the exponent $\nu = 4/7$ (for $d = 2$), which is smaller than for the self-avoiding random walk. An example of a smart-kinetic self-avoiding random walk is pictured in Figure 3.4. Since it is quite hard to get trapped in $d \geq 3$, it is conjectured that this walk behaves the same as the lattice random walk above the upper critical dimension $d = 3$.

An algorithm to create a realization needs to ensure that the walk will not trap itself. One can think of it as a strategy for the game *Snake*.[8] A naive implementation could determine with a simple depth-first search [82] or some heuristically guided search, e.g., A* [83], if a point "outside" is reachable from the planned site of the next step. In $d = 2$ this has a worst case runtime quadratic in the length of the walk.

There is a more clever algorithm [81] requiring only local information which has a constant time complexity per step, i.e., linear in the length of the walk. For this more sophisticated algorithm, we need to save the *winding angle* w_i at every occupied site i. This is just a sum over all turns the walk took, where a left turn is -1 and a right turn is $+1$.

The basic idea is that trapping can only occur if we take a step into a loop. If all sites (a, b, c, d, e) as defined in Figure 3.9 are empty, we can step on any of (a, b, c) and still escape. If some of them are occupied, we need to take a step which does not lead into the loop. To determine which step leads into a loop, we can use the winding angle. If the winding angle on the current site is larger than on the occupied site, the loop is on the right-hand side and vice versa. An example of this case is shown in Figure 3.9(b). Using this, the following algorithm will avoid getting trapped:

---

[8] While the game concept exists since 1976 (*Blockade*), it was popularized by the inclusion on Nokia cellphones since the nineties.

1. Start at x_0 and do a random step
2. Look at the 5 sites in front a, b, c, d, e defined in Figure 3.9 and choose the first fulfilled option
    - if none is occupied, do a random step
    - if exactly one of a, b, c is not occupied, step on it
    - if b is occupied and both a, c are not
        - $w_s > w_b$: step on a
        - $w_s < w_b$: step on c
    - if d and e are not occupied, step on a random not occupied one of a, b, c
    - if d and e are both occupied (a, b, c will all be not occupied)
        - $w_s > w_d > w_e$: step on a
        - $w_d > w_s > w_e$ or $w_e > w_s > w_e$: step on b
        - $w_s < w_d < w_e$: step on c
    - if d is occupied and e is not
        - $w_s > w_d$: step on a
        - $w_s < w_d$: step on c or b
    - if e is occupied and d is not
        - $w_s > w_e$: step on a or b
        - $w_s < w_e$: step on c
3. Repeat 2. until the walk reaches the desired length.

All cases which are not explicitly mentioned are only possible if the walk is already trapped and thus do not occur.

Using this algorithm, it is easy to do simple sampling of realizations. But it is tricky to apply a Markov chain based sampling scheme. The pivot algorithm mentioned above is not applicable, since it would obviously sample the self-avoiding random walk ensemble. Simple changes as used in the lattice random walk can lead to crossings later in the walk, such that one change can result in many necessary changes later to avoid self-crossing. We approach this problem with the black box approach as explained in Section 2.2.3.[9] In contrast to the loop-erased random walk, where this protocol works nicely, the SKSAW is more susceptible to changes in the random numbers $\boldsymbol{\xi}$. Consider that changing a step can shift the whole walk after that point. Therefore, many following steps might be invalid, as they might cross the walk after the shift. Therefore each of those crossing steps would need to be changed, leading to a cascading aggravation of this effect. A replacement of all random numbers ξ_i corresponding to the now invalid steps will introduce too much change to generate atypical realizations

---

[9] The image Figure 2.3 does actually show the evolution of a smart-kinetic self-avoiding random walk.

efficiently. Therefore, we will only change one random number, but interpret the remaining random numbers dependent on the realization, i.e., the random numbers do not determine the direction of their step directly, but are used to choose from the possible directions, i.e., the directions that will neither lead into a trap nor onto an already visited site. While this protocol is still not capable of generating instances over the whole support of the distributions $p(A)$ and $p(L)$, like it is feasible for self-avoiding random walks and loop-erased random walks, we can reach across a reasonable large range and obtain the probability density down to probabilities smaller than 10^{-200}.

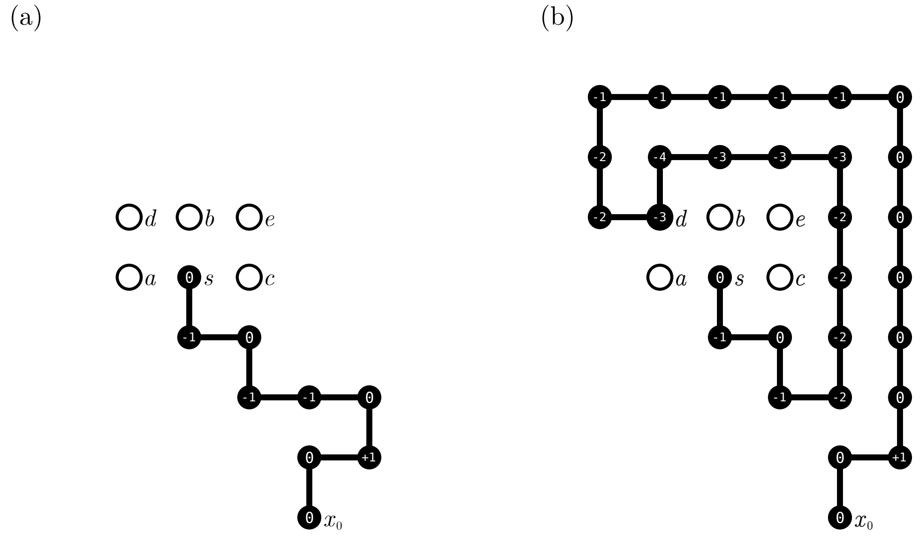


Figure 3.9.: Important concepts for the SKSAW. The head of the walk is labeled as s, its front neighbors a, b, c, d, e are important to determine which step to take next to avoid trapping. The winding angle of every site is written inside the visited sites. (a) A walk where no trapping is possible in the next step. (b) A walk where the a step on any other site than a leads into a trap. The "only d occupied" rule and the winding angle at site s being larger than at site d, i.e., $w_s > w_d$, instructs us to step on site a.

"True" Self-Avoiding Walk

The *"true" self-avoiding random walk* (TSAW),[10] introduced in Reference [84], is different from the self-avoiding walks introduced before, in that it may step on already visited sites, although it is discouraged dependent on an avoidance parameter β. If

---

[10] The reason for the very confident name is, at least partially, that it was one of the first models of self-avoiding walks generated by a growth process, predating the smart-kinetic self-avoiding random walk. The authors wanted to show that the behavior of a growth process, which is intuitively coupled to the "walker" part of the name, is very different from the combinatorial interpretation of the self-avoiding random walk. So in contrast to the self-avoiding random walk, which is more a polymer than a walk, this growth model is a "true" walk.

a site i is visited $n_i(\tau)$ times after τ steps, the "true" self-avoiding random walk will step on site i with probability $p(i, \tau) = e^{-\beta n_i(\tau)}/Z(\tau)$, where

$$Z(\tau) = \sum_{i \in \text{neighbors}} e^{-\beta n_i(\tau)}$$

is just a normalization factor. There are two edge cases of this type of walk. First, $\beta = 0$, where it will not avoid visited sites at all and is identical to the lattice random walk. And second, $\beta \to \infty$ where it will only step on itself, if it is trapped. For large β it behaves therefore, except for the strategy to escape from traps, like the smart-kinetic self-avoiding random walk, which is reflected in the large deviation behavior (cf. Article A.3). A decision tree, where this similarity can also be observed, is shown in Figure 3.6(b). An example for $\beta = 1$ is shown in Figure 3.4.

Since this type is also defined by a growth process, we apply the black box approach to construct the Markov chain. We use for each step one random number to ensure only small changes if one underlying entry of $\boldsymbol{\xi}$ is altered. Technically, of the $2d$ neighboring sites, we step on site $j \leq 2d$, such that j is minimal and fulfilling

$$\xi_k \leq \sum_{i=1}^{j} e^{-\beta n_i}/Z$$

dependent on the random number for the k-th step ξ_k.

In References [84, 85] it was shown that the upper critical dimension of the "true" self-avoiding random walk is $d = 2$, and therefore $\nu = 1/2$ with logarithmic corrections.

Scaling Exponents

As mentioned before, a central quantity to characterize a random walk is the distance typically covered by a walk with T steps. For the one dimensional case it is trivial to calculate. Since the strictly self-avoiding walks[11] can not turn around, obviously $\nu = 1$.

For the lattice random walk the calculation is also straight forward. Let the increments s_i be Rademacher distributed

$$s_i = \begin{cases} 1, & \text{with probability } {}^1\!/\!_2 \\ -1, & \text{with probability } {}^1\!/\!_2. \end{cases}$$

Then the displacement, i.e., the x-coordinate of the end of the walk, is simply

$$x(T) = \sum_{i=1}^{T} s_i.$$

---

[11] The case of the "true" self-avoiding random walk is more involved, and should not be mentioned here. However, the interested reader may read Reference [85] for a derivation of $\nu = 2/3$ in the one-dimensional case.

Since its mean value is zero, one usually looks at the square displacement

$$\begin{aligned} x^2(T) &= \left(\sum_{i=1}^{T} s_i\right)^2 = \sum_{i=1}^{T}\sum_{j=1}^{T} s_i s_j \\ &= \sum_{i=1}^{T} s_i^2 + \sum_{i=1}^{T}\sum_{\substack{j=1\\ j\neq i}}^{T} s_i s_j. \end{aligned}$$

For the last term the following is valid:

$$s_i s_j = \begin{cases} 1, & \text{if } s_i = s_j, \text{which has probability } 1/2 \\ -1, & \text{if } s_i \neq s_j, \text{which has probability } 1/2 \end{cases}$$

such that its mean value is again zero. Considering the first term, with $s_i^2 = 1$ we get

$$\left\langle x^2(T) \right\rangle = \left\langle \sum_{i=1}^{T} s_i^2 \right\rangle = T.$$

Therefore the end-to-end distance R for this case scales as the square root typical for diffusive processes

$$R \propto T^{1/2} \Rightarrow \nu = 1/2.$$

We can generalize this easily to arbitrary dimensions if we replace each of the increments s_i to be Cartesian unit vectors each occuring with probability $1/2d$ and using their orthonormality. Therefore this scaling for the lattice random walk is valid in every dimension.

The self-interacting types are more complicated and have generally larger values for dimensions low enough that the interaction does play a role. To give an intuition why this is the case, one can roughly think of the occupied sites as repelling (in an entropic sense, since there are no forces), such that the walk is pushed away from its past trajectory and thus from the start, resulting in longer distances covered. In higher dimensions the space to explore becomes larger. Thinking of a growth process, it becomes very improbable that the walk will return to an already visited site. The dimension above which the behavior is the same as the standard random walk (according to ν) is called *critical dimension*.

For two dimensions exact values are obtained by non-trivial methods, e.g., based on conformal field theory. For $d = 3$ the self-avoiding random walk and loop-erased random walk proved especially intractable and only numerical estimates exist. Table 3.1 shows known exact values of ν or the best current estimates for ν of all walk types considered in this study.

	$d=1$	$d=2$	$d=3$	$d=4$	$d\geq 5$
LRW [86]	$1/2$	$1/2$	$1/2$	$1/2$	$1/2$
GRW [86]	$1/2$	$1/2$	$1/2$	$1/2$	$1/2$
SAW [75, 87]	1	$3/4$	0.587597(7)	$*1/2$	$1/2$
LERW [88]	1	$4/5$	0.61576(2)	$*1/2$	$1/2$
SKSAW[80]	1	$4/7$	$*1/2$	$1/2$	$1/2$
TSAW [84]	$2/3$	$*1/2$	$1/2$	$1/2$	$1/2$

Table 3.1.: Expected scaling exponents ν for the scaling of the distance between start and end point $R \propto T^{\nu}$. Entries marked by an asterisk $*$ are for the upper critical dimension and subject to possible logarithmic corrections.

3.3.2. Convex Hulls

As already stated in the introduction, the convex hull $\mathcal{C}$ of a set of points $\mathcal{P}$ in d dimensions is the smallest convex polytope including all points of the set $\mathcal{P}$. Convex hulls find wide application in 3D computer graphics and computational geometry, they are so fundamental that the highly cited textbook Reference [89] dedicates two chapters to them. They are a (possible) intermediate step to efficiently calculate *Delaunay triangulations* and their dual *Voronoi diagrams* [90] (to physicists maybe more commonly known as *Wigner-Seitz cell*).

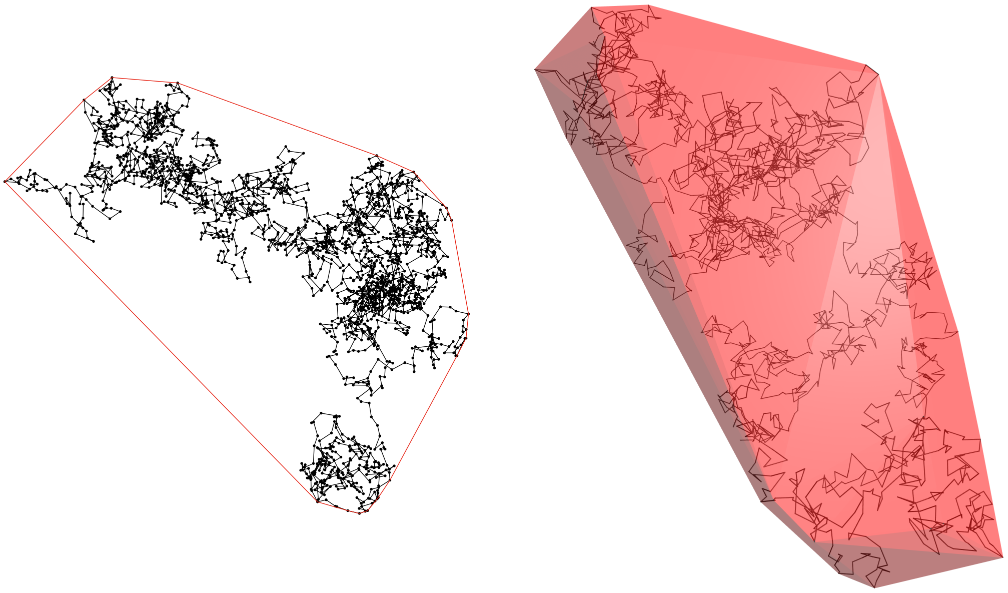


Figure 3.10.: Examples for Gaussian random walks in $d=2$ and $d=3$ with $T=2048$ steps each. Their convex hull is visualized in red.

Note that the number of facets of a convex hull of a random set of points grows exponentially in the dimension. So every exact algorithm to obtain the convex hull, i.e., all of its facets, has an exponential lower bound on its time- and memory complexity $\Omega\left(N^{\lfloor d/2 \rfloor}\right)$ [91–93]. While there are approximation algorithms operating in linear time independent of the dimension [94, 95], we use exact algorithms and are thus limited to rather low dimensions of $d \leq 6$.

There is a wide range of algorithms to determine the convex hull of a given set of points. Since the most time-critical operation in the simulations of this study is the calculation of the convex hull, the available methods need to be evaluated and reviewed carefully. The following subsections will introduce all algorithms used for this study. First some definitions are given to establish a clear language for the geometrical concepts used in the algorithms. A reader already somewhat familiar with computational geometry, might skip this section and use it as a glossary whenever an unknown word in an algorithms' description is encountered. In the following two subsections Andrew's algorithm for convex hulls, which we used for simulations in $d = 2$, and quickhull, which we used for higher dimensions, are explained.[12] We will end this section with methods how to obtain the observables we are interested in. Further, in Appendix B.2.1 a general preprocessing heuristic is introduced and in Appendices B.2.2 and B.2.3 two more exact algorithms, which were considered, are described. In Appendix B.2.4 a comparison of the run times of all implementations considered for the studies of this thesis are given, justifying the choice of algorithms used for the simulations.

Concepts of d-Dimensional Geometry

First, a few definitions will be given. These are mainly notation and necessary for a clear language to describe the following algorithms. Some of the definitions are enriched with technical details for a fast computation of the defined properties, which is crucial for a decent implementation to be able to do extensive simulations. Note that we identify points with the vector pointing from the coordinate origin to their position.

The *hypervolume* V is the generalized volume. In the special case of $d = 2$ it is the same as the area and in $d = 3$ as the volume.

We will call the $(d-1)$-dimensional boundary of a d-dimensional object the *surface* ∂V. In the special case of $d = 2$ it is the same as the perimeter and in $d = 3$ as the surface area.

Given a d-dimensional polytope ($d = 2$: polygon, $d = 3$: polyhedron) its boundary consists of $(d-1)$-dimensional hyperplanes, called *facets* ($d = 2$: edge, $d = 3$: face, $d = 4$: cell). The $(d-2)$-dimensional hyperplanes which are the borders of a facet, are called *ridges* ($d = 2$: vertex, $d = 3$: edge, $d = 4$: face).

The *normal* $\boldsymbol{n}$ of a facet is a normalized vector orthogonal to all pairwise differences of the facet's vertices $\boldsymbol{q_i}$. For intuition in three dimensions, when shifting the normal to the center of the facet, it stands orthogonal on the front of the facet. Which

---

[12] The implementation used to generate the images shown and animations referenced is available at `https://github.com/surt91/convex_hulls`.

side is the front, is conventionally defined by the order of the vertices $\boldsymbol{q_i}$, which are enumerated counter-clockwise when looking on the front of the facet, i.e., the pointing end of its normal. Its direction is straight forward to calculate using, e.g., a generalized crossproduct defined by following determinant:

$$\boldsymbol{q}_1 \times \cdots \times \boldsymbol{q}_{n-1} = \begin{vmatrix} {q_1}^1 & \cdots & {q_1}^n \\ \vdots & \ddots & \vdots \\ {q_{n-1}}^1 & \cdots & {q_{n-1}}^n \\ \boldsymbol{e}_1 & \cdots & \boldsymbol{e}_n \end{vmatrix}, \tag{3.9}$$

where the superscript denotes the component of the vector and $\boldsymbol{e}_i$ are the Cartesian unit vectors. Note that since the entries in the last row are vectors, the result is a vector and not a scalar. The special case of $d = 2$ can be calculated by an extension to three dimensions with a z-component of 0 and crossproduct with $\boldsymbol{e}_z$.

The terms *in front of* and *behind* a facet ($d = 2$: *left of* and *right of*) denote the relative position of a point to a facet considering its orientation. This can be easily calculated given the normal vector $\boldsymbol{n}$ of the facet and some point on the facet $\boldsymbol{q}$, e.g., one of its vertices. Then point $\boldsymbol{p}$ is in front of the facet, if any vector pointing from the facet to $\boldsymbol{p}$ has a parallel component with $\boldsymbol{p}$, i.e.,

$$\begin{cases} p \text{ in front of,} & \text{if } \boldsymbol{n} \cdot (\boldsymbol{p} - \boldsymbol{q}) > 0 \\ p \text{ behind,} & \text{if } \boldsymbol{n} \cdot (\boldsymbol{p} - \boldsymbol{q}) < 0 \\ p \text{ on the same (hyper)plane,} & \text{if } \boldsymbol{n} \cdot (\boldsymbol{p} - \boldsymbol{q}) = 0 \end{cases} \tag{3.10}$$

The reciprocal notation: A facet is called *visible* from a point, if the point is in front of the facet.

By convention all facets of a polytope are oriented such that their normal vectors point to the outside. That means a point $\boldsymbol{p}$ is *inside* a convex polytope, if it is behind all facets, i.e.,

$$\boldsymbol{n}_i \cdot (\boldsymbol{p} - \boldsymbol{q}) < 0 \quad \forall i.$$

Correspondingly a polytope is convex, if none of its facets are visible from any of its vertices.

In two dimensions, we will define the *orientation* o of a triplet of points $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})$ as follows using the z-component of the cross product, again treating two dimensional vectors as three dimensional with a zero z-component:

$$o(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) = \begin{cases} \text{clockwise,} & \text{if } \left[(\boldsymbol{b} - \boldsymbol{a}) \times (\boldsymbol{c} - \boldsymbol{a})\right]_z > 0 \\ \text{counter-clockwise,} & \text{if } \left[(\boldsymbol{b} - \boldsymbol{a}) \times (\boldsymbol{c} - \boldsymbol{a})\right]_z < 0 \\ \text{colinear,} & \text{if } \left[(\boldsymbol{b} - \boldsymbol{a}) \times (\boldsymbol{c} - \boldsymbol{a})\right]_z = 0 \end{cases} \tag{3.11}$$

A d-dimensional simplex ($d = 2$: triangle, $d = 3$ tetrahedron) is a polytope with $d + 1$ vertices, which are linear independent, i.e., a simplex has always a non-zero hypervolume.

Andrew's Monotone Chain

Andrew's monotone chain [96] is a very simple algorithm to determine the convex hull of planar point sets, i.e., it is only applicable in $d = 2$. Its running time for n points $\mathcal{O}(n \log n)$ is dominated by sorting the points.[13]

To be precise, first the input points are sorted according to their x-coordinate and ties are decided by their y-coordinate. This sorting is also called *lexicographical sorting*. In a second step one iterates over the sorted points from left to right.

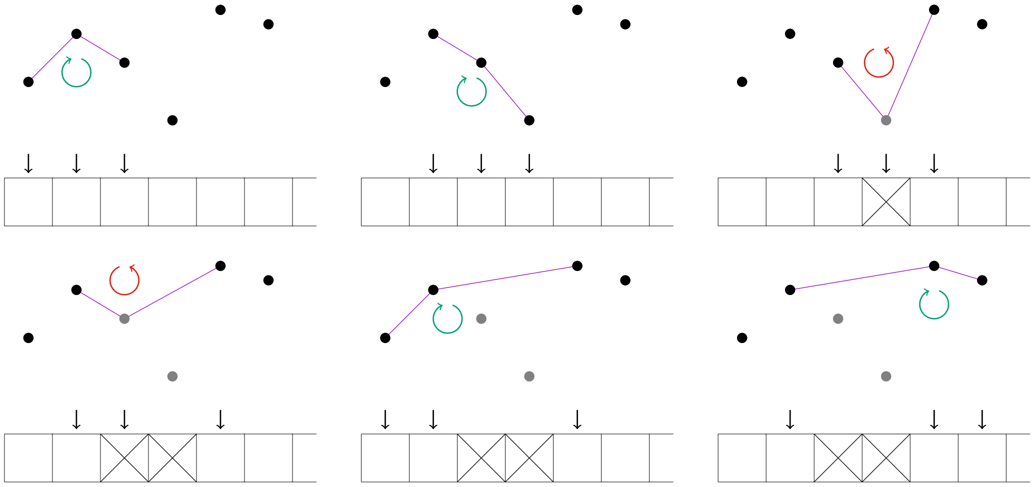


Figure 3.11.: Six steps of Andrew's monotone chain algorithm. The triplets are shown with indicators whether they are clockwise or counter-clockwise. Discarded points are gray. Under the pictures the sorted array of points is visualized. The complete upper hull will be the non-discarded (not marked by crosses) points.

Starting with the three left-most points, we always look at triplets of points. We will identify the triplet i as the point at position i according to the aforementioned sort and the previous two non-discarded points. While the triplet i is counter-clockwise (cf. Equation (3.11)) and there are still enough non-discarded points left, discard the middle point. Then increase i by one. This is repeated until all points were considered, i.e., $i = n$. This process is pictured in Figure 3.11. The non-discarded points form the upper half of the convex hull. Applying the same scheme on the reverse of the sorted points, will yield the lower hull. Since the first point of the upper hull is always the same as the last point of the lower hull and vice versa, they can be merged at these points to get the complete convex hull.[14]

Historically seen, it is an incremental improvement of the algorithms proposed by Graham [98] and later Anderson [99]. The main advantage over the Graham scan, which follows the same idea but sorts according to the polar coordinate instead of the

---

[13] Note that it is therefore possible to reduce the run time to $\mathcal{O}(n)$, if the points were suitable for special sorting algorithms, like *radixsort* [82, 97] for points with integer coordinates in a fixed range.

[14] An animation of this process is available at `https://data.schawe.me/andrew.gif`.

x-coordinate, is the computationally cheaper comparison operation. Note however, that to compare the polar coordinates it is not necessary to calculate trigonometric functions, but it is sufficient to use the above mentioned "is left of" operation as the comparison operation.

Quickhull

Quickhull [100–103] is a divide and conquer algorithm to determine the convex hull. It has an average time complexity of $\mathcal{O}(n \log n)$ in $d = 2$ and $d = 3$ and $\mathcal{O}(n^{\lfloor d/2 \rfloor})$ in all higher dimensions.

The fundamental idea will be first explained in $d = 2$, since it is easy to visualize and understand. Then, the modifications needed for $d = 3$ (and all higher dimensions) are explained.

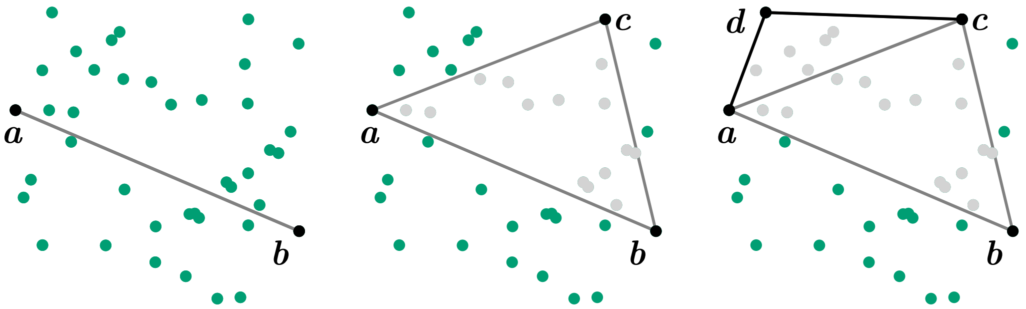


Figure 3.12.: Visualization of the quickhull algorithm with three steps into the recursion. Green points are candidates for members of the convex hull and grey points are discarded.

Start with two points $\boldsymbol{a}, \boldsymbol{b}$ on the convex hull, e.g., the points with minimum and maximum x-coordinate. Determine the point $\boldsymbol{c}$ left of and farthest away from the edge $(\boldsymbol{a}, \boldsymbol{b})$, i.e.,

$$\boldsymbol{c} = \arg\max_{c'} \left\{ \left[(\boldsymbol{c}' - \boldsymbol{a}) \times (\boldsymbol{b} - \boldsymbol{a}) \right]_z \right\}.$$

All points inside the triangle $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})$ can be discarded as they can not be part of the hull. Repeat this step recursively with the edges $(\boldsymbol{a}, \boldsymbol{c})$ and $(\boldsymbol{c}, \boldsymbol{b})$ until there are no points on the left side of the current edge. All edges created in this way on the bottom level of the recursion are part of the convex hull. Two steps of this recursion are pictured in Figure 3.12. The same process is repeated recursively with the point $\boldsymbol{c}'$ left of and farthest away from the inverse edge $(\boldsymbol{b}, \boldsymbol{a})$.[15]

In $d = 3$ we start with a tetrahedron of non-degenerate faces, i.e., faces whose vertices are not colinear. Also its vertices should be extrema in the coordinates such that they surely are part of the convex hull. First, we start with an arbitrary facet f of the initial tetrahedron. Same as in the $d = 2$ case, we need to find the point in front

---

[15] An animation of this process is available at `https://data.schawe.me/quickhull.gif`.

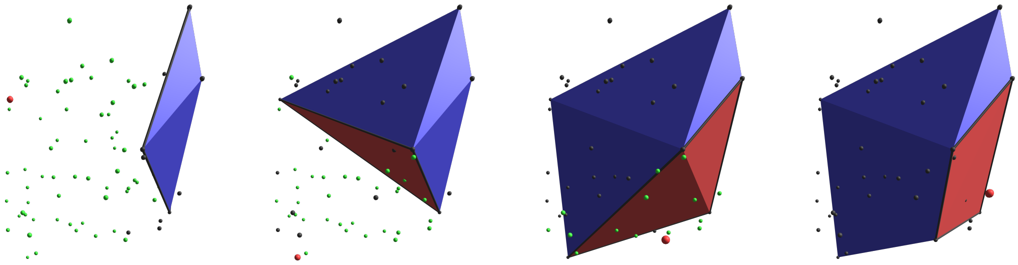


Figure 3.13.: Visualization of the quickhull algorithm in three dimensions with four steps into the recursion. Green points are candidates for members of the convex hull, red is the eye point, the horizon is marked by black lines, blue facets are the intermediate representation of the hull and red facets are visible from the eye point and will be discarded. Shading is just a guide to the eye.

of and farthest away from the facet f. This point is called *eye point* $\boldsymbol{p}$. In contrast to the $d = 2$ case, we now update an intermediate representation of the hull, which is initially the tetrahedron we chose. Next we need to find the *horizon*. It consists of the edges separating visible faces from invisible faces when looking from the point of view of the eye point.[16] Every facet that is visible from the eye point is removed from the intermediate representation and for every edge $(\boldsymbol{u}, \boldsymbol{v})$ of the horizon, the new face $(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{p})$ is added to the intermediate hull. For each of the added facets and the other d facets of the initial tetrahedron, this procedure is repeated recursively until there are no more points in front of any facets. At this time the intermediate representation is the wanted convex hull. A few steps of this recursion are shown in Figure 3.13.[17]

For $d > 3$ there are no more fundamental differences and the above method can be generalized straight forwardly. Nevertheless, the run time increases as $\mathcal{O}(n^{\lfloor d/2 \rfloor})$, since this is the number of facets a convex hull typically contains [91–93], i.e., any algorithm needs at least this time to output every facet defining the hull.

The Volume and Surface of a Convex Polytope

After the construction of the convex hull, we can determine its hypervolume. This is straight forward in $d = 2$ with

$$A = \frac{1}{2} \sum_{i=1}^{|\mathcal{C}|-1} (y_i + y_{i+1})(x_i - x_{i+1}) \tag{3.12}$$

$$L = \sum_{i=1}^{|\mathcal{C}|-1} \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2}, \tag{3.13}$$

---

[16] While it would be possible to test every facet, more efficient methods exists. One possibility is to maintain a graph of neighboring facets and doing a depth-first search starting at f to find the edges separating visible from invisible faces

[17] An animation of this process is available at `https://data.schawe.me/qh3d.mp4`.

with cyclic indices. The sum in Equation (3.12), also known as a variation of the *shoelace formula*, sums the areas of the trapezoids, with parallel edges starting perpendicular on the x-axis and ending in the points y_i and y_{i+1}. This results clearly in the area of the whole convex hull. Even more trivial, the sum Equation (3.13) adds the distances of every consecutive pair of points on the hull.

But in higher dimensions the determination of the volume and surface is a non-trivial problem [104, 105]. While Monte Carlo integration is the only feasible method for general polytopes in high dimensions, in this study we can exploit the convexity of our polytope, since for convex polytopes the hypervolume and surface can be obtained with a recursive scheme.[18] Given the facets f_i, which are $(d-1)$-dimensional simplexes, one can choose an arbitrary fixed point $\boldsymbol{p}$ inside the convex polytope and can create a d-dimensional simplex from each facet f_i, such that their union fills the entire convex hull (cf. Fig. 3.14 for a $d = 2$ example). Therefore the volume can be obtained by summing the volume of all d-dimensional simplexes:

$$V = \sum_i \operatorname{dist}(f_i, \boldsymbol{p}) a_i / d,$$

where $\operatorname{dist}(f, \boldsymbol{p})$ is the perpendicular distance from the facet f_i to the point $\boldsymbol{p}$ and a_i is the surface of the facet. The surface of a $(d-1)$-dimensional facet is its $(d-2)$-dimensional volume, which can be calculated with the same method recursively, until the trivial case of one dimensional facets, i.e., lines. Determining the surface uses the same recursion, by calculating $\partial V = \sum_i a_i$.

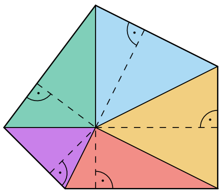


Figure 3.14.: Visualization of the idea to calculate the volume of a convex polygon given its facets and an interior point, perpendicular distances are shown with dashed lines.

To foster intuition, this method is pictured for $d = 2$ in Fig. 3.14. Here, the facets are lines and the volume of the simplex is the area of the triangle. The perpendicular distances are visualized as dashed lines.

3.4. Results

Article A.1 obtains large parts of the distribution of both the (hyper-)volume and surface for Gaussian random walks in dimensions 3 and 4, as well as precise estimates for its variance for long single and multiple walks in up to 6 dimensions. It shows that the same scaling as for the two dimensional case is easily extended to higher dimensions

---

[18] Alternatively Stein's formula [106] can be used $V = \frac{1}{d!} \det(\boldsymbol{v}_1 - \boldsymbol{v}_0, ..., \boldsymbol{v}_d - \boldsymbol{v}_0)$.

and that the numerical data is compatible with this scaling. Further it generalizes a heuristic argument for the limiting form of the rate function (cf. Section 2.1) characterizing the right tail to higher dimensions

$$\Phi(S) \propto S^{2/d_\mathrm{e}},$$

where d_e is the effective dimension of the observable, e.g., $d_\mathrm{e} = 2$ for the surface area of a $d = 3$ dimensional hull. Also for the left tail behavior we could observe behavior which we expected, due to the close relation to the support function M (cf. Section 3.1.1).

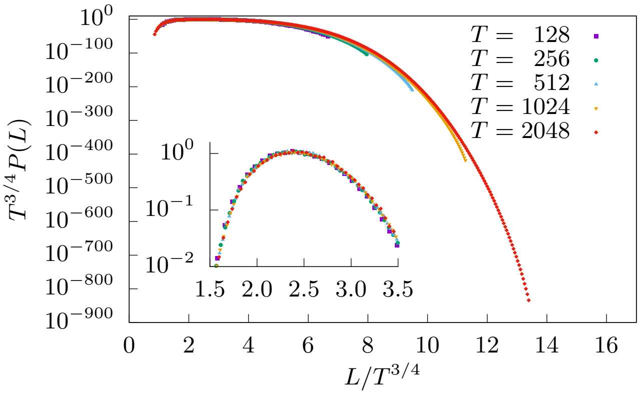


Figure 3.15.: Demonstration of the scaling of the distribution of the perimeter L of the self-avoiding random walk as derived in Article A.2 for different sizes T.

Article A.2 approaches the lack of results for non-Markovian random walks, i.e., random walks which have a memory. It studies a selection of three models for self-avoiding walks, namely the self-avoiding random walk, the loop-erased random walk and the smart-kinetic self-avoiding random walk. For all three the same generalized scaling of the whole distribution parametrized by the growth exponent ν of the walk type and the dimension was proposed and supported by numerical data down into the deep tails of very low probabilities. An example of such a scaling, which was evaluated for the publication Article A.2 but not shown therein, is shown in Figure 3.15. Here a collapse, i.e., the independence of the system size after the scaling except for finite-size effects, is well visible. This is especially interesting since the behavior of the distribution's tails is therefore only dependent on observables which can be obtained from a study of the high-probability region. The numerical large deviation data was used to underpin an asymptotic functional form derived by an heuristic argument for the right-tail of the rate function. This is a generalization of the relation observed in previous studies (cf. Section 2.1 and Article A.1) dependent on ν

$$\Phi(S) \propto S^{1/d_\mathrm{e}(1-\nu)}, \tag{3.14}$$

where d_e is the effective dimension of the observable.

Article A.3 handles the “true” self-avoiding random walk, an ensemble of self-avoiding walks, whose self-avoidance can be tuned with a parameter β. In particular at the extreme $\beta = 0$ it reduces to a standard random walk. The first surprising result of this study was that at the $\beta \to \infty$ extreme the “true” self-avoiding random walk behaves the same as the smart-kinetic self-avoiding random walk in the far right tail, but differently, especially with a different growth exponent ν, in the main, i.e., high probability, region. This was the first example where a collapse using the observables obtained from the main region does not reflect the behavior of the large deviation tails. Correspondingly the estimate for the rate function Equation (3.14) does not work, but the same form with the growth exponent ν of the smart-kinetic self-avoiding random walk was observed.

4. Groundstate Energy of a Generalized Random Energy Model

In this chapter we will look at a problem which is of fundamental interest for pure mathematics and additionally has a physical application as a simple toy model for non-interacting fermions. We will first motivate it from both the mathematical and physical side and shed a bit of light on the background of a very similar and famous physical toy model in Section 4.1. Afterwards we formulate our research question in Section 4.2. In Section 4.3 the technical details for an efficient implementation of the model in the context of our large deviation sampling approach is explained in minute detail. Consecutively Section 4.4 gives an overview over the results we obtained in Article A.4.

4.1. Current State of Research

The background of our research problem can be motivated from a purely mathematical point of view as well as from a physical point of view. We will start with the mathematical viewpoint, as it nicely connects two quite fundamental concepts.

The first is the central limit theorem, which was already shortly mentioned in Section 2.1. It says that the sum of N identically, independently distributed (i.i.d.) random numbers with finite mean and variance will converge to a Gaussian distribution in with increasing N. This is one of the most fundamental and well known theorems of stochastics. It is the fundament of error analyses in many experimental disciplines, as the most basic error estimates assume Gaussian errors, which is a sensible guess because of the central limit theorem.

Similarly, there are theorems about the distribution of the maxima of N i.i.d. random numbers which are collected under the term extreme-value theory. If the N random variables Q_i are i.i.d. according to the distribution p and if p decays exponentially, the distribution of the maxima will converge to a Gumbel distribution $P(x \leq \max_i\{Q_i\}) = \mathrm{e}^{-\mathrm{e}^{-(x-\mu)/\beta}}$ with parameters μ and β dependent on the details of p. Also for the case of slow decaying distributions, the distribution of the maxima will, under mild assumptions, converge to a Fréchet distribution and for the case of a support with an upper bound to a Weibull distribution.

Now we ask a generalized question, which includes these two theorems as limiting cases.

> What is the distribution of the sum S of the K largest values of N i.i.d. from p drawn random numbers in the limit of $N \to \infty$?

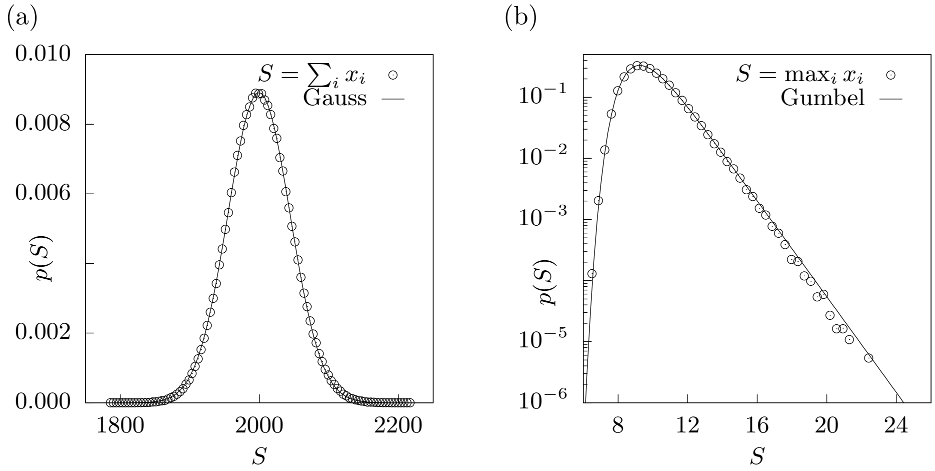


Figure 4.1.: $N = 1000$ random numbers x_i distributed according to the Erlang distribution $p(x) = x\mathrm{e}^{-x}, x \geq 0$. (a) Estimate for the distribution of their sum ($K = N$). (b) Estimate for the distribution of their maximum ($K = 1$). Apparently (a) behaves like a Gaussian in the main region and (b) like a Gumbel distribution. The data were obtained by simple sampling over 10^6 realizations.

For the case $K = 1$ the answer is one of the extreme value distributions dependent on the type of the distribution p, for the case $K = N$ it is the Gaussian distribution. Both extreme cases are visualized in Figure 4.1. For K in between these two extrema the distribution is exactly known if p is an exponential distribution [107]. For this case a comparison to our large deviation simulations shows a good agreement with the exact result over a very wide range, which is visualized in Figure 4.2. For general distributions p this is still an open question.

The direct connection of this mathematical problem to a physical problem is a “toy” quantum system with N energy levels and K non-interacting fermions. In the ground state the K particles will occupy the K lowest energy levels and the system will therefore have a total energy E_0 equal to the sum of the K smallest values. The energy distribution is therefore exactly the above introduced problem if we look at the minima instead of the maxima. The extreme value distributions are still applicable for minima after a trivial change of variables $x \to -x$.

Interestingly, this model is in structure very similar to the *random-energy model* introduced in Derrida’s highly cited work [108, 109]. The random-energy model consists of independent random energy levels, much like the above introduced model. The main distinctions are that energy levels of the random-energy model are typically Gaussian, while in our model we look at arbitrary distributions with positive support. Anyway, it is instructive to look into the background of the random-energy model.

One of the most interesting properties of the random-energy model is that one can construct a Hamiltonian, which connects it to the iconic *Sherrington-Kirkpatrick* (*SK*) mean field model [110] for spin glasses [111, 112]. Therefore consider a system of N

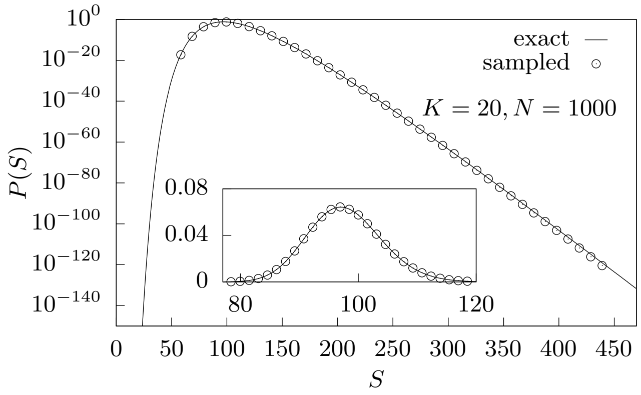


Figure 4.2.: Distribution of the sum of the largest K values of N exponentially distributed random numbers. This data were preliminary tests and are not published in Article A.4. The inset shows the same data in linear scale. (For clarity not every data point is visualized.)

interacting Ising spins[1] σ_i. Let

$$\mathcal{H}_{\mathcal{P}}(\{\sigma_{\mathcal{P}}\}) = - \sum_{\{i_1,\dots,i_{\mathcal{P}}\}} A_{i_i i_2 \dots i_{\mathcal{P}}} \sigma_{i_1} \dots \sigma_{i_{\mathcal{P}}} \tag{4.1}$$

be the Hamiltonian, where the sum goes over all groups of $\mathcal{P}$ spins $\{i_1, \dots, i_{\mathcal{P}}\}$ and each group has an individual interaction term $A_{i_i i_2 \cdots i_{\mathcal{P}}}$. The interactions are chosen as suitably normalized Gaussians. Obviously, for $\mathcal{P} = 1$ we have a system of N free spins without interaction. For $\mathcal{P} = 2$ we have interaction of all pairs of spins, which is the Hamiltonian of the *SK* spin glass model. For the limit of large $\mathcal{P}$ Reference [108] shows that the 2^N energy levels ε_i become independent Gaussian distributed. This enables the analytical study of this model in contrast to spin glasses, where correlations complicate the calculations.

Derrida derives the most interesting thermodynamic properties, like the partition function and the free energy, of the random energy model in his seminal papers [108, 109] and shows rich critical behavior. There is a phase transition below a finite critical temperature from an unordered state to a frozen state, where the system is always in its ground state. Further, adding an additional pairwise interaction to this model

$$\mathcal{H}_{\mathcal{P}}'(\{\sigma_{\mathcal{P}}\}) = \mathcal{H}_{\mathcal{P}} - \frac{J_0}{N} \sum_{\langle ij \rangle} \sigma_i \sigma_j \tag{4.2}$$

leads to the classical para- and ferromagnetic phases. Additionally, there are two frozen states, one without magnetization at low pairwise coupling J_0 and the other with finite magnetization at larger pairwise coupling.

---

[1] Since it would take too much space to introduce the Ising model properly and most readers will already be familiar with the so called “Drosophila of statistical mechanics,” I will not go into detail here. However, a reader who would like to read my words explaining it, can either read my bachelor’s thesis [22] or Reference [20].

This model was never intended to describe a real physical system, but rather to pose a toy model simple enough to teach, understand and deploy new methods. In this regard, it has to be considered a success. In the history of spin glasses, it was used, e.g., to test the plausibility of Parisi's *replica symmetry breaking* [113, 114] ansatz, where in Reference [115] a solution obtained using the replica approach for the general $\mathcal{P}$ case could be shown to coincide in the limit $\mathcal{P} \to \infty$ with the exactly known properties of the random energy model. Also, Talagrand's rigorous treatment of spin glasses in Reference [116] starts with a chapter about the random energy model.

4.2. Research Question

As mentioned above, our model is very similar to Derrida's random energy model, though we are free to choose any distribution p for the energy levels ε_i. We do not want to study the finite temperature behavior like Derrida has, but are only interested in the ground state, corresponding to the frozen phase of Derrida's original model. Correspondingly, the question we want to answer is

> What is the distribution $P_K(E_0)$ of the groundstate energy E_0 of the K smallest energy levels ε_i of N random energy levels i.i.d. according to an arbitrary distribution p?

Or in the mathematical framework introduced in the beginning:

> What is the distribution $P_K(S)$ of the sum S of the K smallest values of N i.i.d. from p drawn random numbers?

We tackled this question in Article A.4. From both an analytical point of view in the limit of $N \to \infty$ and with numerical simulations for finite N, which show a convergence to the analytically obtained asymptotic form.

4.3. Models and Methods

While the model is quite simple and can be stated in only a few lines, we will spend a bit more time with the numerical methods used to examine it in the paragraphs that follow. For the details of generating random numbers for the different distributions studied in Article A.4, the reader is referred to Appendix B.3.

The model we are scrutinizing is defined by a distribution of the energy levels $p(\varepsilon)$, the number of energy levels N and the number of fermions K. We name the energy levels by ascending energy, i.e., $\varepsilon_1 \leq \ldots \leq \varepsilon_N$. Naturally, each energy level can only be occupied by one fermion, such that the ground state energy of the system is

$$E_0 = \sum_{i=0}^{K} \varepsilon_i. \tag{4.3}$$

A visualization of an example realization is shown in Figure 4.3.

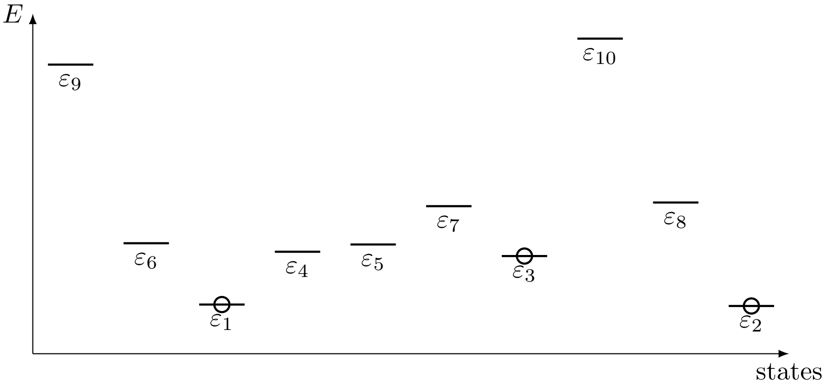


Figure 4.3.: Visualization of a groundstate of our random energy model. There are $N = 10$ energy levels ε_i of which the $K = 3$ lowest are occupied. Here, the energy levels are distributed according to the distribution $p(\varepsilon) = \varepsilon e^{-\varepsilon}, \varepsilon \geq 0$.

Efficient Implementation Since extensive numerical simulations profit from an efficient implementation, we will dedicate here more space for the implementation details than would be appropriate in an article. The naive implementation of an MCMC change move for this model would change a random energy level ε_i to ε', sort the sequence in time $\mathcal{O}(N \log N)$ anew and sum the the first K values of the newly sorted sequence in $\mathcal{O}(K) = \mathcal{O}(1)$ (because K is fixed). Obviously we do not have to sort the whole sequence from scratch if only one energy level is changed. In fact, we can first remove the ε_i to be changed from the sequence and exploit the order of the remaining sequence to insert the new energy level ε' at the correct place. Here, we have to go a bit into technical details of the implementation for an optimal strategy.

One option would be storing the sequence in an array. The advantage is that selecting the element ε_i to be changed is possible in $\mathcal{O}(1)$ using an index. Finding the correct position of the new value ε' can be done in $\mathcal{O}(\log N)$ using binary search. The disadvantage is that the removal and insertion operation both take time $\mathcal{O}(N)$, since on average half of the entries need to be copied to new positions in the memory.

The next option would be using a binary tree for the sequence. This way the order is automatically maintained during insertions and removals. Also both operations only take time $\mathcal{O}(\log N)$ in the worst case, when we use a self-balancing version of a binary tree, like a *red-black tree* [82] or an *AVL tree* [82]. But there is no easy way to select a random element from a tree. One would have to read $\mathcal{O}(N)$ elements to be able to uniformly draw one random element.

So we have two approaches both resulting in $\mathcal{O}(N)$ time complexity for one evaluation of the energy. Fortunately, we can combine both data structures to achieve a time complexity of $\mathcal{O}(\log N)$ per iteration. We maintain an unsorted array of the energy levels ε_i to draw random entries in constant time. Therefore we draw an index i of the unsorted array uniformly. The candidate for the change move $\varepsilon^{(i)}$ at position i can then be found in, erased from and its replacement ε' inserted into the binary

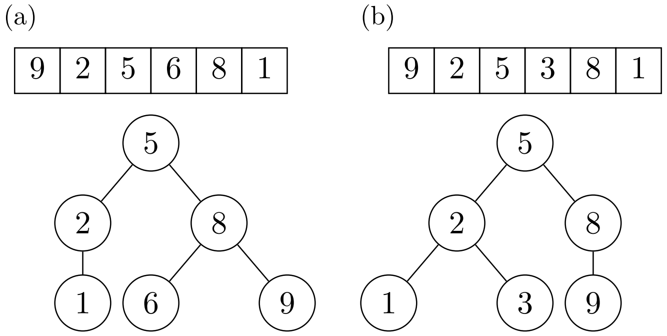


Figure 4.4.: Visualization of the data structure used for the simulation of our random-energy model. (a) before the change, (b) after the change. The entry at position 4 in the unsorted array is selected by chance and its value determined via the array as $\varepsilon^{(4)} = 6$. Consequently 6 is removed from the tree. A new random energy level $\varepsilon' = 3$ is proposed and written to the position 4 in the array. Also, $\varepsilon' = 3$ is inserted into the tree. The new energy E_0 can now be obtained by traversing and summing the first K elements ε_i of the tree.

tree in $\mathcal{O}(\log N)$.[2] Also we have to overwrite the entry at position i in the array with the replacement value ε'. The new energy can be calculated in $\mathcal{O}(\log N)$ in the worst case by traversing and summing the first K elements in the tree. If this change move is rejected, the operation to undo it is in the same complexity class. A small example is shown in Figure 4.4.

Markov Chain Change Move With the above techniques it is trivial to generate a typical realization of the ensemble. Just drawing N random numbers as energy levels ε_i, sorting them and summing the smallest K of the random numbers results in a uniform sample for E_0. But constructing a Markov chain to perform importance sampling using the black box approach (cf. Section 2.2.3) uncovers some difficulties. Especially for the extreme case of large E_0, it is clear that every single energy level ε_i needs to be large. If we just replace a random entry of the underlying vector $\boldsymbol{\xi}$ with a new uniform random number, the new entry will be smaller than the one it replaces in most cases. This change will lead to a smaller E_0 than before and will therefore most likely be rejected, if we are aiming for the large E_0 region. To mitigate this problem, we refined the black box approach by not replacing a chosen ξ_i but changing it slightly. We still have to ensure that we sample the correct statistics, i.e., when always accepting the change moves, the entries ξ_i need to maintain a uniform distribution. Therefore we change a chosen entry ξ_i to $\xi' = \xi_i + \delta\eta$, where $\eta \in [-1, 1]$ is uniformly distributed and $\delta \in \{10^{-i} | i \in \{0, 1, 2, 3, 4, 5\}\}$ also uniform determines the scale of the change. In the case $\xi' \notin [0, 1)$, the proposal is automatically rejected. Note that this protocol only works if the ξ_i are uniform random numbers, thus we generate the energy levels

---

[2] Mind that we could also store pointers to the nodes of the tree in the array, which would give us constant time find and removal. But since insertion still takes $\mathcal{O}(\log N)$ time, we will stay here at this easier formulation where every value exists twice.

ε_i following the distribution $p(\varepsilon)$ of interest in every iteration from the underlying uniform random numbers ξ_i (or in the case of Gaussian from two uniform random numbers ξ_{2i} and ξ_{2i+1}). The exact methods for this are shown in Appendix B.3.

For our extreme example this means that at large values of E_0, where every value of ε_i is large, due to the construction of these ε_i from the underlying ξ_i, every ξ_i is either very close to 0 or very close to 1. Now there is a high probability to change them only very slightly on the scale of, e.g., 10^{-5}, such that the corresponding ε_i stays large and may be accepted.

This refined method enables us to sample deeper into the right tail than possible before. Also this protocol is still generally applicable to any problem, if similar problems should be encountered.

4.4. Results

In Article A.4 we find a large N asymptotic form $F_K^{(\alpha)}(z)$ for the distribution $P_K(E_0)$ with fixed K. This asymptotic form is universal for distributions whose underlying $p(\varepsilon)$, decays fast enough, has positive support and behaves as $p(\varepsilon) \approx B\varepsilon^\alpha$ for small ε. The rescaled argument is $z = bE_0N^{1/(1+\alpha)}$ with $b = (B/(\alpha+1))^{1/(\alpha+1)}$.

The following paragraphs will be a "tourists guide" through the paper. The analytical part of the paper is the contribution of my coauthors, Grégory Schehr[3] and Satya Majumdar. Without going into too much detail, the basic approach of the analytical derivation should be sketched.

First, an explicit joint probability distribution for the K smallest random numbers

$$P(\varepsilon_1, \cdots, \varepsilon_K) = \frac{\Gamma(N+1)}{\Gamma(N-K+1)} \prod_{i=1}^{K} p(\varepsilon_i) \prod_{i=2}^{K} \Theta(\varepsilon_i - \varepsilon_{i-1}) \left[\int_{\varepsilon_K}^{\infty} p(u)\, \mathrm{d}u\right]^{N-K} \tag{4.4}$$

is derived using combinatorial arguments. Basically, the combinatorial factor in front is the number of ways we can draw K numbers from a pool of N numbers, the first product calculates the probability to draw the values of the energy levels ε_i and the second product calculates the probability that the first K entries of the random selection are ordered using the Heaviside function Θ. The integral calculates the probability for the remaining $N-K$ numbers to be larger than ε_K. Starting from this the probability for a certain energy E_0 of the system to occur is a simple K-dimensional integral over the product of a Dirac δ function and $P(\varepsilon_1, \cdots, \varepsilon_K)$. This way the probabilities of every configuration resulting in the system energy E_0 are summed leading to a complicated expression for $P(E_0)$. The structure of $P(E_0)$ lends itself to perform a Laplace transform on it for simplification, such that from there on the calculation operates on the distribution's Laplace transform $\left\langle e^{-sE_0} \right\rangle$, a concept we already encountered in Section 2.1 and Equation (2.7). The Laplace transform can be simplified using approximations for large N. At this step the influence of the distribution $p(\varepsilon)$ from which the

---

[3] In September 2016 I spent a month at the same institute as Grégory, the LPTMS in Orsay and in the summer 2017 I attended the FPSP XIV summer school where Grégory was a lecturer.

random numbers are drawn is reduced to one parameter α, which is determined by the behavior of $p(\varepsilon) \sim \varepsilon^{\alpha}$ for small ε. The result will therefore not only be applicable to special distributions $p(\varepsilon)$, but is universal for whole classes of distributions $p(\varepsilon)$. This results in an expression for the Laplace transform of the asymptotic distribution $F_K^{(\alpha)}$:

$$\int_0^\infty F_K^{(\alpha)}(z)\mathrm{e}^{-\lambda z}\,\mathrm{d}z = \frac{(\alpha+1)^K}{\Gamma(K)\lambda^{(\alpha+1)(K-1)}}\int_0^\infty x^{\alpha}\mathrm{e}^{-\lambda x - x^{\alpha+1}}\left[\gamma(\alpha+1,\lambda x)\right]^{K-1}\,\mathrm{d}x\ , \tag{4.5}$$

with the incomplete gamma function $\gamma(a,x) = \int_0^x \mathrm{d}u\, u^{a-1}\mathrm{e}^{-u}$. For the special case of $\alpha = 0$ the transform can be inverted yielding an elementary expression for the asymptotical distribution of interest

$$F_K^{(0)}(z) = \sum_{n=1}^{K}(-1)^{K-n}\frac{n^{K-1}}{(K-n)!\,n!}\,\mathrm{e}^{-z/n}\ . \tag{4.6}$$

This special case includes exponential, half-Gaussian and Pareto distributed ε. Indeed we show for these three distributions numerically the convergence to the same universal limiting form $F_K^{(0)}(z)$. Interestingly, the convergence happens from different directions. The exponential case shows very little to no deviations from the limiting form at finite sizes N. The Pareto case is always larger than the limiting form for finite N and the Gaussian case is always lower than the limiting form. All those results are pictured in Figure 1 of Article A.4.

Values extrapolated point-wise with the ansatz for finite-size corrections of

$$P_{K,N}(E_0) \approx b\,N^{1/(1+\alpha)}\left[F_K^{(\alpha)}(z) + N^{-\beta}G_K^{(\alpha)}(z) + N^{-2\beta}H_K^{(\alpha)}(z)\right]\ , \tag{4.7}$$

with the correction terms $G_K^{(\alpha)}(z)$ and $H_K^{(\alpha)}(z)$ assumed as constants and obtained by a least squares fit nicely match the asymptotic form $F_K^{(0)}(z)$.

For the general case $\alpha \neq 0$ the expression for the Laplace transform of $F_K^{(\alpha)}(z)$ in Equation (4.5) is simple enough for numerical inversion and is used to compare to numerical results. However, we need to employ a multiprecision library because of the precision requirements to numerically inverse Laplace transforms, especially since the precision of the result needs to be exceptionally good as our obtained distributions are precise down to 10^{-160}. We therefore use the implementation of de Hoog's algorithm [117] provided by the multiprecision library `mpmath` [118].[4]

Using the same procedure, we also show a convincing convergence to the analytical result for $p(\varepsilon) = \varepsilon\mathrm{e}^{-\varepsilon}, \varepsilon \geq 0$ corresponding to $\alpha = 1$. This can be seen in Figure 4.5.

---

[4] `mpmath` implements advanced algorithms on the primitives provided by `GMP`, the GNU multiprecision library.

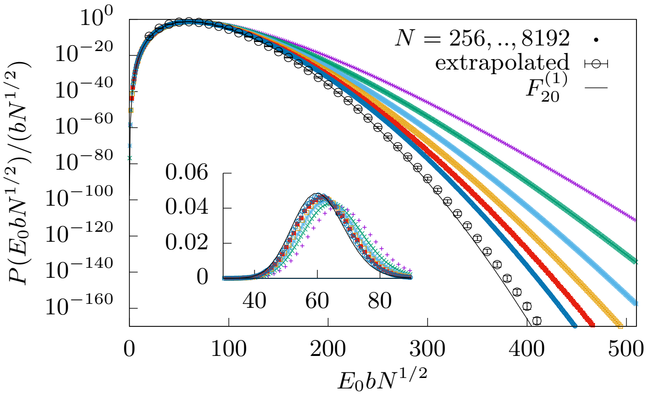


Figure 4.5.: Distribution of the sum E_0 of the K smallest values of N random numbers distributed according to $p(\varepsilon) = \varepsilon e^{-\varepsilon}, \varepsilon \geq 0$ rescaled and extrapolated to match the asymptotic expectation. This figure is taken from Article A.4.

5. The Longest Increasing Subsequence

The *longest increasing subsequence* (*LIS*) seems to find its first mention as an example to demonstrate Monte Carlo sampling in the text book Reference [119]. The corresponding chapter was written by Stanisław Ulam, who we remember as one of the founding fathers of the Monte Carlo method from Section 2.2, and therefore this problem is also sometimes called *Ulam's problem.* Given a sequence S with N entries s is a *subsequence* of S if all entries of s also occur in S in the same order, but not necessarily without gaps. As a visual clarification, one can obtain a subsequence by removing arbitrary elements from a sequence, e.g., $s = (\cancel{3}, 9, \cancel{4}, \cancel{1}, 2, 7, \cancel{6}, \cancel{8}, \cancel{0}, 5) = (9, 2, 7, 5)$. If the elements of s are strictly increasing, it is called *increasing subsequence.* The *LIS* is now an optimization problem to find the longest of all subsequences, such that the entries are strictly increasing. Note that the LIS is not necessarily unique, e.g., in the following sequence one LIS is marked with underlines and one marked with overlines: $S = (\underline{3}, 9, \underline{4}, \overline{1}, \overline{2}, \underline{7}, \overline{6}, \underline{\overline{8}}, 0, 5)$. The length L of the LIS, in this example $L = 4$, is our observable of interest. In his chapter of Reference [119] Ulam uses simple sampling to estimate the expected length $\langle L \rangle$ of the LIS of random permutations. He suggests a linear relation of $\langle L \rangle \approx 1.7\sqrt{N}$, which is today known exactly in the limit of long sequences to be $\langle L \rangle = 2\sqrt{N}$ [120].

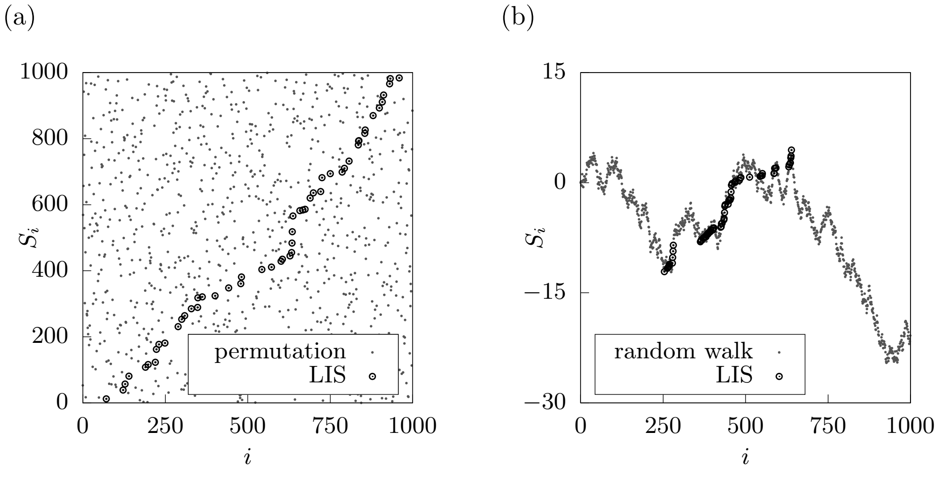


Figure 5.1.: (a) Random permutation S_i of all integers $0 \leq j < 1000$ plotted at their corresponding position i in the sequence. The $L = 53$ entries constituting one of the LISs are marked with circles. (b) Random walk S_i with steps from a uniform distribution $U(-1, 1)$. The $L = 66$ entries constituting one of the LISs are marked with circles.

In the context of this thesis, we are interested in more than just mean values and indeed explicit expressions for limiting forms of the rate function, each valid in one of the tails, are already known [121–123]. Also the asymptotic limit of the whole distribution after suitable rescaling is known [124] to converge to a *Tracy-Widom distribution* [125].

To appreciate this result, we have to take a look at the Tracy-Widom distribution.[1] While there is no closed form representation for its probability density function, it is a universal distribution appearing in seemingly unrelated problems. The first problem in which the Tracy-Widom distribution was spotted originates from random matrix theory: For specific ensembles of random matrices the largest eigenvalue is distributed according to the Tracy-Widom distribution, in the limit of large matrices. More precisely, there are mainly three Tracy-Widom distributions of interest, which are labeled by a parameter $\beta \in \{1, 2, 4\}$. While technically, there are Tracy-Widom distributions for other values of β, only these three values have a nice interpretation from random matrix theory. Since random matrix theory is far beyond the scope of this thesis, we shall only look shallowly at the ensemble corresponding to $\beta = 2$, the *Gaussian unitary ensemble*.[2] This ensemble of random matrices consists of Hermitian matrices M of size $N \times N$, i.e., the entries are complex numbers M_{ij}, such that they are equal to their complex conjugate after transposition, i.e., $M_{ij} = \overline{M_{ji}}$. Since the matrix is Hermitian, the diagonal elements $M_{ii} = a_{ii}$ are real i.i.d. Gaussians with zero mean and a variance of $1/2$. The nondiagonal elements are defined as $M_{ij} = \overline{M_{ji}} = a_{ij} + \mathrm{i}b_{ij}$, where a_{ij} and b_{ij} are i.i.d. Gaussians with zero mean and variance of $1/4$. The probability density function of this ensemble is given as $P(M)\,\mathrm{d}M = \frac{1}{Z}\mathrm{e}^{-\beta/2\,\mathrm{Tr}\,M^2}\,\mathrm{d}M$, where Z is a normalization constant, $\beta = 2$ and $\mathrm{d}M = \prod_{i=1}^{N} \mathrm{d}a_{ii} \prod_{i<j} \mathrm{d}a_{ij}\,\mathrm{d}b_{ij}$.

Finding the Tracy-Widom distribution known from random matrix theory in the statistics of the seemingly unrelated LIS hints at a kind of universality of this distribution. This universality was confirmed when it was found in growth processes. At this time the Tracy-Widom distribution became very interesting for physicists. There are for example mappings of very simple growth models to the LIS [127–130] showing the connection clearly. Apparently the universality of the Tracy-Widom distribution is connected to the KPZ universality class originating from the Kardar-Parisi-Zhang (KPZ) equation describing the fluctuations of the height in growth processes. This growth can also be observed in experiments, like those conducted by Kazumasa Takeuchi [131, 132]. In these experiments, a liquid crystal is grown and the growth is captured by a camera. From the images the height h of the growth at time t is captured and suitably rescaled $\chi = (h - v_\infty) + (\Gamma t)^{1/3}$, with two constants v_∞ and Γ. This setup enabled Takeuchi to gather enough data to observe directly that the

---

[1] I had the opportunity to listen to a lecture of Kazumasa Takeuchi [126] at the summer school FPSP XIV about the *KPZ* equation, Tracy-Widom distribution and a primer of random matrix theory. This section therefore borrows the notation and examples from my handwritten notes of this lecture [7].

[2] The name stems from the single entries being Gaussian distributed and the eigenvalues being invariant under unitary transforms due to their Hermitian structure. The other two ensembles are the *Gaussian orthogonal ensemble* ($\beta = 1$) and *Gaussian symplectic ensemble* ($\beta = 4$).

distribution of the measured values χ follows the Tracy-Widom distribution for probabilities as small as 10^{-4}. Indeed, when starting from a line, this results in the $\beta = 1$ Tracy-Widom distribution (corresponding to the Gaussian orthogonal ensemble) and when starting from a point and observing circular growth it results in the $\beta = 2$ Tracy-Widom distribution (corresponding to the Gaussian unitary ensemble).

Now that we understand that the relevance of the LIS is not only in this combinatorial problem itself, but rather in the connection of this very simple model to a whole class of problems – from random matrix theory to the growth of surfaces described by the KPZ equation – we will review the research literature about the LIS in more detail in Section 5.1. In Section 5.2 we will formulate our research question. In Section 5.3 we will introduce the ensembles whose LIS we scrutinized and the methods, which are needed to examine our research question. Finally in Section 5.4 the results obtained during our study are shortly summarized.

5.1. Current State of Research

The first study, we will look at in more detail, is about the mapping of a surface growth model, for which it is sensible to assume that it falls in the KPZ universality, to the LIS. In Reference [129] an easy anisotropic *ballistic deposition* growth model is introduced. Note that this is not the first growth model for which a mapping to LIS was found. The first one was a time-continuous *polynuclear growth* model introduced in Reference [127]. The advantage of the ballistic deposition model is that it is discrete in time and space and therefore easier to visualize and in turn easier to understand, thus we will look at this later model.

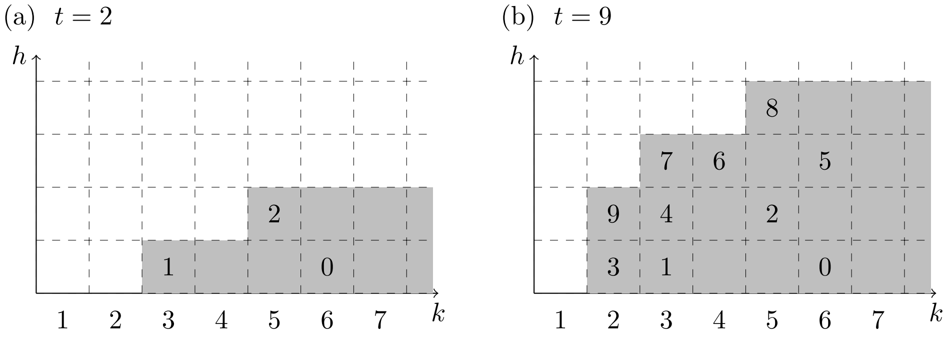


Figure 5.2.: Anisotropic ballistic deposition on a lattice of size $l = 7$ with ten timesteps. Ten blocks are falling sequentially on the positions $6, 3, 5, 2, 3, 6, 4, 3, 5, 2$. The numbers indicate the timestep in which this block was dropped. Gray sites are blocked either by a block or because they are right of a block. This process can be mapped to the sequence $S = (3, 9, 4, 1, 2, 7, 6, 8, 0, 5)$ using the formalism from Reference [129]. One LIS is in this example $\mathrm{LIS}(S) = (1, 2, 6, 8)$, with length $L = 4$, which is also the height at the highest point.

The growth model is an anisotropic ballistic deposition model in $1 + 1$ dimensions. Ballistic deposition generally means that entities with some size are moving on straight, i.e., *ballistic*, lines and stick to other entities on contact after which their position is fixed for the remainder of the process. In the model of interest here, in each time step a *block* is dropped on a random site drawn uniformly from all sites of a one dimensional lattice with l sites. On contact, it extends all the way to the right, but not to the left. In following timesteps new blocks may fall on the blocked sites and thus increase the height. A few snapshots of this process are pictured in Figure 5.2. Using an ingenious mapping [129] the height of the pile of blocks at site k and at time t can then directly be mapped to the length of the LIS of a random permutation of N numbers, where N is a random variable with the distribution

$$P_{k,t}(N) = \binom{t}{N} \left(\frac{k}{l}\right)^N \left(1 - \frac{k}{l}\right)^{t-N} . \tag{5.1}$$

Especially, the height at the highest point, i.e., at $k = l$, is equal to the length of a LIS of a random permutation of $N = t$ distinct numbers. Since the mapping itself is not trivial and is very well explained in Reference [129], we will not look further into the mapping.

So if we want to study the height of of this growth process, we can instead study the length L of LISs of random permutations, for which fortunately the whole asymptotic distribution is known already to be a Tracy-Widom distribution [124]. This can be used with suitable rescaling to predict the height h_k of the k-th column on a lattice with l sites at time t of the ballistic deposition process for the limit of large values of l and t and a fixed ratio tk/l

$$h_k(t) \to 2\sqrt{\frac{tk}{l}} + \left(\frac{tk}{l}\right)^{1/6} \chi, \tag{5.2}$$

where the random variable χ is distributed according to the Tracy-Widom distribution. This allows the authors to arrive at the asymptotic average height $\langle h_k(t) \rangle = 2\sqrt{tk/l} - \langle \chi \rangle (tk/l)^{1/6}$ and asymptotic variance $\sigma_k^2(t) \to \langle [\chi - \langle \chi \rangle]^2 \rangle (tk/l)^{1/3}$.

There are also numerical studies concerning the LIS. Quite recently Ricardo Mendonça[3] published an article about what happens to the distribution if the sequences are not random permutations, but random walks [133]. The fact that something changes is easy to see in Figure 5.1. Since the entries of the sequence generated by a random walk are strongly correlated, we can observe long runs with an upward or downward trend. Naturally, the LIS is then confined to one of the upward trends and consists typically of more elements. Previous mathematical work suggested that the mean value of the length $\langle L \rangle$ should scale like $\langle L \rangle \approx cN^\theta$ with a positive constant c. Especially, for random walks with steps drawn from a distribution with finite variance the exponent is expected to be $\theta = 0.5$ with possible logarithmic corrections [134]. For steps drawn from distributions with diverging variance, there are rigorous bounds for a limiting case of extremely heavy tailed distributions [135], which limit the exponent to

---

[3] During my time at the LPTMS, I listened to a talk of Ricardo Mendonça and talked to Satya Majumdar about this topic, which inspired this study.

a value inside the bounds $0.690.. \leq \theta \leq 0.815...$ To be more precise this result applies to α-stable distributions with α "sufficiently small" [135]. A distribution is said to be α-stable if, loosely speaking, the distribution of the sum of n i.i.d. samples from this distribution behaves as $cn^{1/\alpha}$ times the distribution itself [136] with $\alpha > 0$ and some constant $c > 0$. For example the standard normal distribution is α-stable with $\alpha = 2$ and for $\alpha < 2$ the variance diverges. The lower the value of α, the more heavy the tail of the distribution. If α is not small enough, the corresponding exponent θ characterizing the length of the LIS is conjectured to interpolate between this case and the $\theta = 0.5$ exponent known for the LIS of random walks with steps of finite variance.

In Reference [133] this expected scaling behavior of the mean and the high probability part of the distribution is studied on a wide range of one dimensional random walk models using simple sampling. This way evidence is gathered that for step-length distributions with diverging variance, for which α-stable distributions with $\alpha \in \{1/2, 1, 3/2, 7/4\}$ were used, the lower bound $0.690.. \leq \theta$ is approached from below when approaching the limiting case $\alpha \to 0$ for which the bound was derived, supporting the above stated conjecture. For each of the different types of random walks the distribution of the length $p(L)$ is rescaled according to the scaling assumption

$$p(L) = N^{-\theta} g\left(N^{-\theta} L\right) \tag{5.3}$$

yielding convincing collapses on the not explicitly known scaling function g, which indicates that this scaling is not only valid for the mean values but also for the high probability regions of the distribution. Indeed their data for random walks with step length of finite variance is good enough that they give a conjecture for the constants and the logarithmic correction of the scaling for walks with increments of finite variance as

$$\langle L \rangle \approx \frac{1}{e}\sqrt{N} \ln N + \frac{1}{2}\sqrt{N}. \tag{5.4}$$

5.2. Research Question

Since the testing of known results is a fundamental ingredient for any science, we would like to first reproduce the Tracy-Widom distribution for the LIS of random permutations, though we want to determine it far more precisely than before, i.e., we want to study also the large deviation regime at probabilities smaller than, say, 10^{-100}. While the asymptotic form is known to be a Tracy-Widom distribution, it is not known how fast the convergence is and if we can observe this shape for finite system sizes. At the same time, we want to gather data to test the conjectures of Reference [133], which showed a convincing scaling form of the distribution of the length L in the high probability region. We want to test whether the same scaling is applicable also in the tails of the distribution. Our research question will be formulated as follows.

> Is a convergence to the asymptotically known form of the distribution of the lengths L of LISs of random permutations visible for finite system

> sizes also in the far tails? And do the conjectures tested on the main region of the distribution of L of LISs of random walks also hold in the far tail? What do the far tails look like?

These questions motivated a research project, whose results are shown in Article A.5.

5.3. Models and Methods

Since the LIS is already well defined in the first paragraph of this chapter, we will in this section only introduce the ensembles of sequences for which we studied the length of the LIS and in more detail than necessary for this study the algorithm we used to determine this length.

We studied two ensembles of sequences. First random permutations of integers, which is the best studied ensemble. A random permutation of N distinct integers is a configuration drawn uniformly from the set of all $N!$ possible orderings of those integers. The generation of a random permutation is possible in time $\mathcal{O}(N)$ by shuffling an arbitrary sequence of the elements with (a variation) of the Fisher-Yates algorithm [137]. As a change move to construct a Markov chain of permutations, we simply exchange two random entries. Note that this ensemble has the same properties as an often studied two dimensional LIS variant as shown in, e.g., Reference [123].

The second ensemble consists of one dimensional random walks with step lengths from a uniform distribution $U(-1, 1)$. They are already defined in great detail in Section 3.3.1, such that we will just rename the entities of Equation (3.8): The sequences of this ensemble $S_i \equiv x(i)$.

5.3.1. Patience Sort

Since we are not interested in the LIS itself but only its length, we can use a peculiar property of the *patience sort* algorithm. While it was originally designed as a sorting algorithm, its connection to LISs was also found early [138]. Interestingly, it was recently rediscovered as a sorting algorithm especially suited for partially sorted data [139]. Its basic idea is to sort the N elements into L sorted stacks, which can then be combined into the sorted sequence similar to *merge sort* [82]. The crucial property of patience sort for our purposes is that the minimal number of sorted stacks is equal to the length of the LIS of the original sequence [140].

To arrive at the minimal number of stacks, a greedy algorithm is sufficient [140]. Since this sorting algorithm is named after a game of cards – *patience* is just the British name for *solitaire* – we will explain this process with a deck of cards. Though, for clarity we will identify each card with an integer instead of the classical symbols. Starting with a sequence S, we imagine that every entry is a card on our hands, with the first entry on top of the deck. We take the top card from our deck and open up a new stack with it. For the next card on top of our deck we go from left to right through every stack already on the table and place this card on the first stack which shows a value greater than our current card. If we do not find such a stack, we open

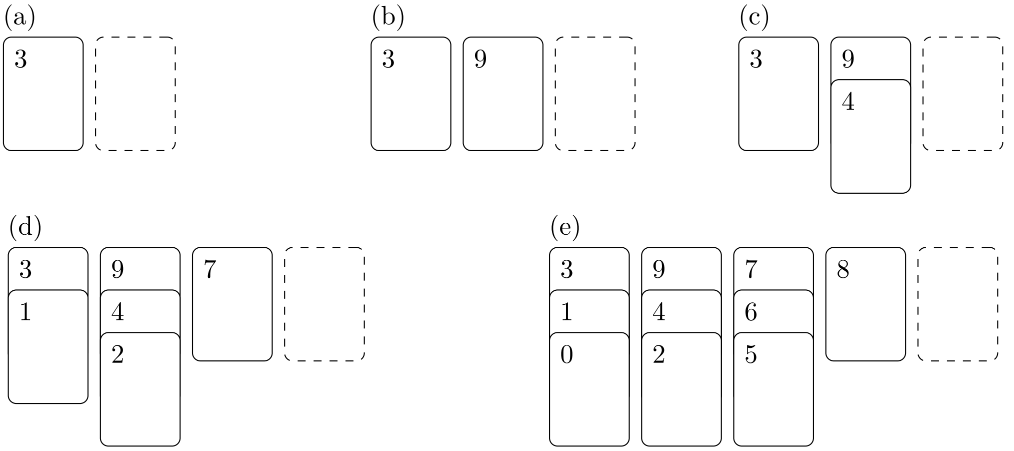


Figure 5.3.: Visualization of patience sort using stacks of cards. The input sequence is $S = (3, 9, 4, 1, 2, 7, 6, 8, 0, 5)$. The single panels show snapshots at different times. In (e) all cards are used and there are $L = 4$ stacks, i.e., the length of the LIS is $L = 4$.

up a new stack right of the last one. Repeating this procedure with every card of our deck, will result in L stacks of cards which are sorted. To finish the sorting, we can always take the card with the least value (like an L-way mergesort), which will always be the top card of one stack, and arrive at a sorted deck.

However, we are only interested in the number of stacks L, thus we can simplify the algorithm somewhat. First, we only ever need the top card of each stack, such that it is sufficient to only save the topmost card instead of the whole stack. Second, the top cards are always ordered due to construction, such that we do not have to go from left to right and test every card in $\mathcal{O}(N)$ (consider the worst case, where the deck is already sorted and N stacks are needed), but we can rather use binary search to find the leftmost stack greater than our card in time $\mathcal{O}(\ln N)$. Since this search has to be performed once for each card, we arrive at a time complexity of $\mathcal{O}(N \ln N)$.

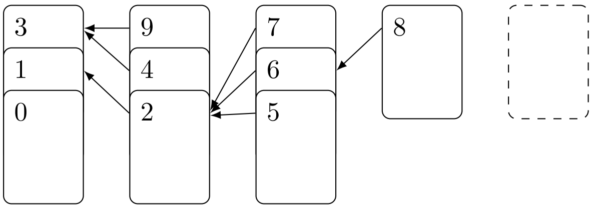


Figure 5.4.: Visualization of the backpointer extension of patience sort to obtain a LIS instead of only its length. The sequence is the same as in Figure 5.3 $S = (3, 9, 4, 1, 2, 7, 6, 8, 0, 5)$. A reversed LIS can be read off following the pointers starting at the last stack, $8 \to 6 \to 2 \to 1$, resulting in $s = (1, 2, 6, 8)$.

Note that we can not obtain an actual LIS this way but only its length. However, going beyond what is needed for Article A.5, we can modify the algorithm slightly to obtain an actual LIS. This modification is rather instructive, as it makes it easy to understand why the number of stacks is equal to the length L of the LIS. Every time we put a card on a stack, we also save a pointer to the top card of the previous stack on the card, as shown in Figure 5.5. This predecessor card is by construction smaller than the current card and occurred earlier in the sequence. In the end we can follow the backpointers from any card of the last stack to obtain the reverse of a LIS s. Since the backpointers will always point to a neighboring pile, the length of s is equal to the number of stacks [140].

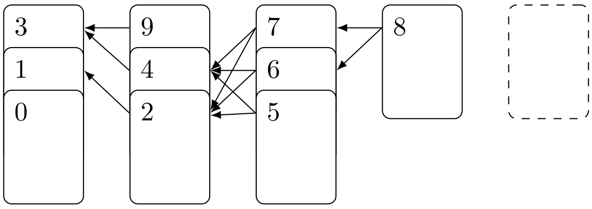


Figure 5.5.: Visualization of a modification of the backpointer extension of patience sort to obtain all LIS instead of only one. The sequence is the same as in Figure 5.3 $S = (3, 9, 4, 1, 2, 7, 6, 8, 0, 5)$. All four reversed LISs can be read off following the pointers starting at the last stack, resulting in $s_1 = (1, 2, 6, 8), s_2 = (1, 2, 7, 8), s_3 = (3, 4, 6, 8), s_4 = (3, 4, 7, 8)$.

We can again slightly extend this structure to not only read off one LIS but all LISs. Therefore we construct a *directed acyclic graph*, i.e., a directed graph without cycles (for definitions on the graph terminology see Section 6.1), to construct all LISs. At each step we do not only store one pointer to the previous top card, but to every card on the previous stack smaller than the current card. Every LIS will be encoded as a traversal of this directed acyclic graph from one of the nodes included in the last stack to a leaf. It should even be possible, using a dynamic programming approach, to count the number of distinct LIS in an efficient way using this structure.

5.4. Results

In Article A.5 we compare the full distribution to the analytically known rate functions [122, 123] for the left tail

$$\lim_{n\to\infty} \frac{1}{N} \ln P(L) = -2H_0\left(L/\sqrt{N}\right) \tag{5.5}$$

and right tail [121, 123]

$$\lim_{N\to\infty} \frac{1}{\sqrt{N}} \ln P\left(L\right) = -U_0\left(L/\sqrt{N}\right). \tag{5.6}$$

Note that in contrast to the general form introduced in Equation (2.1) in Section 2.1, the right-tail behavior behaves as $P(L) \propto \mathrm{e}^{-\sqrt{N}U_0(L)}$, i.e., a leading $\sqrt{N}$ term. For the rate functions $2H_0$ and U_0 closed form expressions exist. Using our distributions at finite sizes, we can confirm clearly the convergence of the rescaled distributions to the rate functions with increasing N in Figure 5.6.

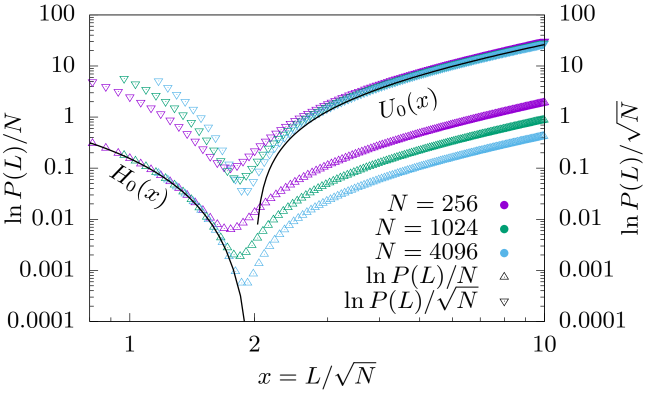


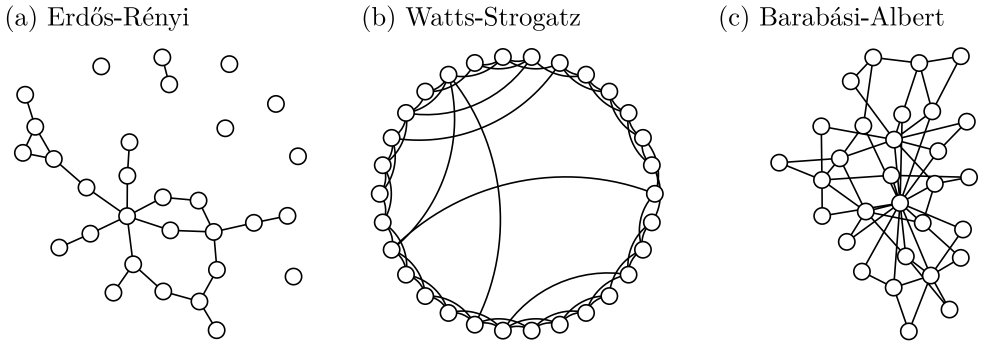
Figure 5.6.: Distribution $P(L)$ of the length of the longest increasing subsequence of random permutations. On the top rescaled (cf. right y-axis), such that it collapses on the right-tail rate function U_0, on the bottom rescaled (cf. left y-axis) to collapse on the left-tail rate function. A similar plot is shown in Article A.5.

For the LIS of random walks, we gathered data of basically the whole distribution, especially containing data for the edge case $L = N$ coinciding with the expected probability of all increments being positive, i.e., $P(L = N) = 2^{-N}$. For the behavior of the rate function Φ, we can estimate the leading order exponents to be $\Phi_l(L) \sim L^{-1.5}$ for the left tail and $\Phi_r(L) \sim L^{2.9}$ for the right tail. These are clearly different from the random permutation case (left: $H_0(L) \sim L^{-3}$, right: $U_0(L) \sim L^{3/2}$).

6. The Largest Biconnected Component of Random Graphs

In the most abstract formulation *graphs* are mathematical objects describing relations between other objects, which makes them useful in a wide variety of applications. Graphs were long known and used in a mathematical context. The founding myth is Euler's approach to solve the *Königsberger Brückenproblem*, a riddle whether, given the topology of the city Königsberg, it was possible to take a walk crossing every bridge exactly once.[1]

Notably, the branch of *random graph theory* was founded in Reference [141] by Paul Erdős and Alfréd Rényi in 1960, where they introduced and studied an ensemble of random graphs. The graphs they studied, of which one example is shown in Figure 6.1(a), are named Erdős-Rényi graphs after them and will be introduced in more detail in Section 6.4.2.


(a) Erdős-Rényi

(b) Watts-Strogatz

(c) Barabási-Albert

Figure 6.1.: Visualization of different types of random graphs. (a) Erdős-Rényi graph with $N = 30$ nodes and $M = 25$ edges. (b) Watts-Strogatz graph with $N = 30$ nodes and a rewiring probability of $p = 0.1$. (c) Barabási-Albert graph where in every iteration a node with $m = 2$ incident edges is added to the graph until there are $N = 30$ nodes.

In physics graphs became a very productive field around the year 2000 as the world became more connected and computers and therefore data collection became ubiquitous. At this time large amounts of data about networks, e.g., transportation networks, the internet, social networks or protein interaction networks were collected and the study of those networks showed that they had different properties in comparison to

---

[1] While it was impossible during Euler's lifetime, after World War II two bridges were destroyed, such that it is possible at time of writing.

existing ensembles of random graphs [142–144]. This prompted the invention of random graph ensembles capturing these new properties and allowing to study them with mathematical machinery.

There are roughly two classes of random graph models on which the literature focuses. On the one hand are graphs whose nodes have a typical degree, i.e., number of incident edges, close to the mean degree $k \approx \langle k \rangle$ and nodes with extremely high degrees are statistically insignificant. Their degree distribution could be, e.g., Poissonian. The Erdős-Rényi graph is the most studied model of this class.

Another influential model of this class is the *Watts-Strogatz* graph [145] as shown in Figure 6.1(b). This random graph model addresses the *small-world* phenomenon, i.e., a property many social networks have, which typically show many local relations but also some long reaching shortcuts. While the local relations are not very surprising, since your friends are also often friends with each other, the long shortcuts lead to unexpected behavior. The most famous experiment in this regard was in the 1960s and is known as “six degrees of separation” [146]. Consider a letter without address but only a name. To send it to the intended recipient, which is a random person, you send it to someone you know, i.e., which is a neighbor of you in the social graph. This process needs allegedly typically only six iterations before it reaches the intended recipient.[2]

On the other hand there are *scale-free* graphs, which seem to be ubiquitous in nature and show surprisingly different behavior. Their defining characteristic is that their degree distribution follows a power law $p_k \propto k^{-\gamma}$. That means that there is no typical degree which determines the behavior, but the distribution of the degrees is heavy tailed and nodes with very high degree are statistically significant. Often, the variance or even the mean of the degree distribution are diverging depending on the exponent γ. A selection of networks from very different backgrounds whose degree distribution follows at least over a few decades a power law is shown in Figure 6.2.

The most influential model for this class is probably the *Barabási-Albert graph*. It was introduced in Reference [144], about 20 years after the first random graph model using a similar mechanism leading to scale-free graphs was introduced in Reference [150]. An example Barabási-Albert graph is pictured in Figure 6.1(c). Scale-free networks capture this universal property of many networks observed in reality, like scientific citation networks or other social networks, as well as protein interaction networks.

In scale-free networks, due to the heavy tailed degree distribution, there are few hubs with very high degree and many nodes with very low degree. This leads to unexpected properties concerning the robustness of those networks. While a disease, e.g., described by the SIR model [66], which was already shortly introduced in Section 3.1, has to be infectious beyond some threshold to survive, this threshold is zero on scale-free networks [68]. On the other hand, scale-free networks are robust against random node

---

[2] Similarly there are “six degrees of Bacon” for the small world phenomenon in the movie industry and the “Erdős number” for the scientific community (the Erdős number of the author of the thesis at hand is 4 at the time of writing).

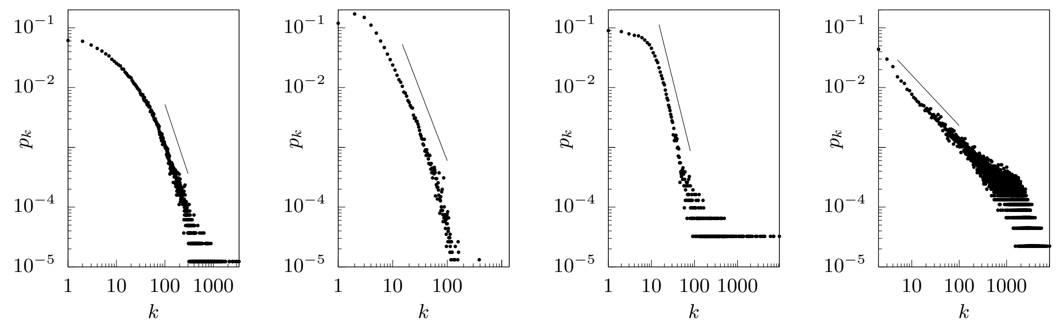

(a) Twitter (b) Citeseer (c) Linux (d) Mouse Genome

Figure 6.2.: Degree distribution p_k for a selection of real world networks. Each follows a power law over a few decades hinting at their scale free nature. (a) Degree distribution of the social network of some Twitter users, each node symbolizes a user and each edge a “follow” relationship [147]. (b) Degree distribution of a citation network according to data of the scientific search service *Citeseer* [148]. (c) Degree distribution of the network consisting of the source files of the Linux project, each node is a source file and the edges show which files are `#include`d [149]. (d) Degree distribution of a gene regulatory network of a mouse genome [148].

failure, explaining the robustness of, e.g., the internet [151].

Naturally, robustness is a very important property for networks. Especially when designing networks, like power grids or communication networks, it is vital that the functionality of the whole network is not compromised by the unavoidable malfunction of single components. Sometimes robustness is an undesirable property, e.g., for disease spreading. So to limit the spreading of disease among cattle, it would be worthwhile to study the weak points of the current topology and use this knowledge to limit the functionality of the network for disease spreading with minimal effort.

The remainder of this chapter will give a quick overview over the field of complex networks with a focus on the robustness of networks in Section 6.2, before we will formulate our research question in Section 6.3. Section 6.4 will give an explanation of the models and algorithms used to arrive at the results of Article A.6, which will be shortly summarized in Section 6.5. But first we will quickly introduce some definitions in Section 6.1 to have a clear language for the following sections.

6.1. Definitions

Graphs are defined as the tuple $G = (V, E)$. The *node set* V, contains the objects, the *edge set* E contains their relation. The relations can be either *directed* $(u, v) = e \in E \subset V \times V$ or *undirected* $\{u, v\} = e \in E \subset V^{(2)}$ and may be weighted with a weight w_e. Edges connecting a node with itself $e = \{u, u\}$ do not occur in simple graphs. The *size* of a graph is the number of nodes $N = |V|$ and the number of edges is denoted by $|E|$. Since Article A.6 does only study undirected and unweighted graphs, we will limit this chapter also only to this type.

Two nodes u and v are *neighbors* if $\{u, v\} \in E$. The edges $\{\{u, v\} \mid \{u, v\} \in E \; \forall v \in V\}$ are called *incident edges* to u. The number of incident edges of u is its *degree* k. The *degree distribution* of a graph p_k is the probability that a node has the degree k.

A *path* is a sequence of edges between two nodes u and v, such that two consecutive edges of the path always share a node, i.e., $(\{u, k_1\}, \{k_1, k_2\}, \ldots, \{k_i, v\})$. A path starting and ending in the same node is called *cycle*. Two nodes between which a path exists are called *connected*. The maximal sets of nodes which are pairwise connected are called *connected components*. A *subgraph* G' is a part of another graph $G = (V, E)$, to be more precise $G' = (V', E')$ with $V' \subset V$ and $E' \subset E$. A *tree* is a graph in which each pair of nodes is connected by exactly one path.

The maximal sets of nodes between which pairwise two *node-independent* paths exist, i.e., two distinct paths whose edges do not share any node except the start and the end, are called *biconnected components*. The maximal sets of nodes between which pairwise two *edge-independent* paths exist, i.e., two distinct paths which do not have any common edges, are called *bi-edge-connected components*. The remainder of a graph from which all nodes with degree less or equal q are iteratively removed, is called *q-core*.

6.2. Current State of Research

One of the seminal papers founding the field "Physics of Complex Networks" is Reference [144], where Albert-László Barabási and Réka Albert show that a growth process, where nodes are iteratively added preferentially to nodes with a high degree,[3] leads to graphs with a degree distribution following a power law $p(k) \propto k^{-3}$. Since growth with *preferential attachment* is plausibly a very common process, it becomes understandable that scale-free networks appear in a wide range of contexts [152], natural as well as cultural, as the selection in Figure 6.2 confirms. Note however that there might be other processes also leading to scale-free networks.

An important property distinguishing scale-free networks from networks with a non-heavy-tailed degree distribution is their robustness to failures. Reference [153] conducts a study about the tolerance against errors and targeted attacks of Erdős-Rényi graphs and Barabási-Albert graphs. It measures the functionality of the network as the *diameter*, i.e., the longest of all shortest paths connecting all pairs of nodes. They show that random errors, i.e., the removal of random nodes, in Erdős-Rényi graphs have a larger impact on the functionality than random errors in Barabási-Albert graphs. This is easy to understand since every node in an Erdős-Rényi graph is roughly equal, while in a Barabási-Albert graph most nodes are only connected to very few other nodes and have no influence on the functionality as no shortest path traverses them. Interestingly, scale-free graphs seem to become susceptible to targeted attacks. The diameter of a Barabási-Albert graph grows fast as the nodes with the highest degree,

---

[3] Note that this is the same idea used in the much earlier published article Reference [150], where it is called *cumulative advantage*.

which act as hubs, are removed. Erdős-Rényi graphs are not susceptible to these targeted attacks, since there are no especially important nodes.

Similar studies on the robustness of networks were carried out. Some focusing on the networks underlying our civilization, like power grids [154–156] or the internet [151, 157]. Some working on more general or more abstract graph models [153, 158–160]. This plethora of robustness studies uses many different observables to determine the functionality of the network. While some of the more involved studies on power grids model each node as producers and consumers influencing the frequency of the AC current in the network, the more fundamental studies look at simpler observables, like the size of the largest connected component [158] or the size of the largest biconnected component [159].

We shall now look at bit closer at the latter study Reference [159]. Here biconnected components are used as a proxy for the robustness of a network. A biconnected component is a connected component which stays connected if any member node is removed This way it is very intuitive that a network with a large biconnected component is very robust against single node failures. Reference [159] takes an analytic approach to express the mean size of the largest biconnected component exactly in the large N limit given the degree distribution p_k of the graph. This derivation assumes that the connections of the nodes of the graph are independent and is therefore applicable to Erdős-Rényi graphs and the configuration model [161], a graph ensemble drawing uniformly from every graph with a given degree distribution p_k. We will here spend some time to understand their reasoning as it fosters the understanding of the observable we are interested in and their results will directly motivate our research question.

There are two cases, which have to be handled separately. Either there is a *giant connected component*, i.e., a connected subgraph of size $\mathcal{O}(N)$, or there is none. For the case that there is none, the uncorrelated nature of the ensembles in question leads to almost all connected subgraphs being trees[4] of size $\mathcal{O}(1)$, and since the size of the largest connected component is always larger than the size of the largest biconnected component, the expected relative size of the largest biconnected component vanishes for large graphs, i.e., $\langle S_2 \rangle = 0$.

Thus, we will now look at the more interesting case, where a giant connected component exists. By definition, two nodes belong to the same biconnected component if they are connected via two paths which have no common node, i.e., node-independent paths. We can approximate this criterion with an upper bound if we ignore that the two paths need to be node-independent. Since in the limit of large graphs the probability that the paths are not node-independent vanishes [162], this approximation becomes exact in the large N limit. Under this approximation a node is part of the biconnected component if at least two of its neighbors are part of the giant compo-

---

[4] As a qualitative argument, consider that there are $\mathcal{O}(N)$ connected subgraphs of size $\mathcal{O}(1)$. To form a biconnected component there has to exist an edge originating at one connected component and connecting to the same connected component, which has a probability proportional to the relative size of this component, i.e., $\mathcal{O}(1/N)$. Combining this with the number of connected components leads to a number of biconnected components of $\mathcal{O}(1)$, i.e., they are extremely rare for large graphs [159].

nent, since there will be paths across the giant component to the giant biconnected component. Therefore we only need to study the probability of a node to be part of the giant component. Let u be the probability that a node is *not* part of the giant component, then Reference [163] shows a self-consistent equation to determine u. The idea is that the chance not to be in the connected component is equal to the chance that none of the neighbors are part of the connected component. First, we have to introduce the *excess degree distribution* q_k. This distribution describes the degree of a random neighbor w of a node v, excluding the edge back to the node v. It differs from the degree distribution p_k of random nodes, since the probability to reach node w when following a random edge of node v is proportional to the degree of w.[5] More precisely,

$$q_k = \frac{(k+1)\,p_{k+1}}{\langle k \rangle}, \tag{6.1}$$

where the $+1$ terms account for the edge back to v and the denominator is for normalization [163]. Now consider the probability u that if we follow an edge we will arrive at a node outside of the giant component. Since this node has k^* neighbors (excluding the node we originate from), all neighbors must not be part of the giant component, which is the case with probability u^{k^*}. To arrive at an expectation value, we have to average this over the distribution of the excess degree

$$u = \sum_{k=0}^{\infty} q_k u^k, \tag{6.2}$$

which we can solve – at least numerically – for u given the degree distribution p_k.

Knowing what the probability is for a neighbor to not belong to the giant component and knowing that for $N \to \infty$ a node will be part of the giant biconnected component if it has two or more neighbors in the giant component, we arrive at

$$S_2 = 1 - \sum_k p_k u^k - \sum_k p_k k (1-u) u^{k-1}. \tag{6.3}$$

The second term is the expected value for a node to have not a single neighbor in the giant component and the third term is the expected probability for a node to have exactly one neighbor in the giant component. One minus their sum is the expected value of a node being in the giant biconnected component, which is at the same time the expected relative size of the giant biconnected component. Note that $1 - \sum_k p_k u^k$ is the size of the giant component. Since there can not be a giant bicomponent without a giant component, the two will always occur at the same time, except for the special case of $1 - \sum_k p_k u^k = \sum_k p_k k(1-u)u^{k-1}$.

Further, the authors generalize this result to m-connected components, i.e., with m node-independent paths, which also all arise at the same threshold value, though the

---

[5] This effect leads in social networks to the friendship paradox: Your friends have typically more friends than you have [164].

transition is of order $m+1$ for these kind of graphs. That means that the growth of the m-connected components with increasing connectivity is slower the larger m is.

Using this result we could now calculate the size of the biconnected component for any Erdős-Rényi graph or configuration model ensemble, at least numerically. As an example for an Erdős-Rényi graph where each edge occurs with probability $p = 2/N$, the degree distribution is known to be $p_k = \binom{N-1}{k} p^k (1-p)^{N-1-k}$. Using this to solve Equation (6.2) numerically and calculate Equation (6.3) results in a behavior converging to to the value $S_2 = 0.473...$

Next to these analytical results, there are a few papers looking at the distribution of similarly simple observables in random graphs. Namely Reference [31] looks at the distribution of the size of the giant component S including the very rare event tails and compares the numerical results to an analytically known rate function (cf. Section 2.1), which is known for Erdős-Rényi graphs above the percolation threshold [165]. The results of the simulations in that study match the analytically known rate function remarkably well, even for the finite sizes at which the simulations were done. More interestingly, very recently Reference [37] was published, which looked at the full distribution of the size of the 2-core $S_{\text{2-core}}$. More general, the q-core is the remainder of a graph after iteratively removing every node of degree less than q. This makes it suited as a simple observable for the robustness of a network [166]. Consider a model of components which fail, if they are connected to less than q other components, e.g., a simplified electrical network, where current can only flow through a node if it has a source and a drain. The 2-core of this network will be the part of the network with current flow. In Reference [37] it was found that the general shape of the distribution $P(S_{\text{2-core}})$ has a shape reminiscent of the distribution of the size of the largest connected component S. Though, they are not similar as they can not be collapsed by a simple rescaling of the axes.

6.3. Research Question

Now that we understand the analytical derivation of the mean value of the size of the largest biconnected component, one obvious question is for higher moments or even the whole distribution. Since there is a numerical study of another simple criterion for robustness, namely the 2-core, we should compare these two observables. Also it would be interesting to look into the qualitative differences when probing this quantity on two fundamentally different graph ensembles. Therefore we formulate our research question:

> What is the distribution, including its low probability tails, of the size of the largest biconnected component of Erdős-Rényi graphs and Barabási-Albert graphs? Does a rate function exist and how does it compare to other observables used for robustness?

This question inspired work, whose numerical results are shown in Article A.6.

6.4. Models and Methods

6.4.1. Finding Biconnected Components

An efficient algorithm to identify all biconnected components is formulated by Hopcroft and Tarjan in Reference [167]. The basic idea is to use a *depth-first search* to explore the graph [82]. The depth-first search works by starting with a stack containing an arbitrary node, putting all its unmarked neighbors on the stack, marking them and removing the entry for the node itself from the stack. Each time a new node is put on the stack, the process is interrupted and started on the new node, leading to the *depth-first* part of the name. Clearly, at any time there is a path from the top node of the stack to any other node on the stack. To determine the biconnected components, we have to ensure a second independent path, i.e., it may traverse only nodes which are not currently on the stack. This can be efficiently done by storing some additional information at each node: the *depth* of the search and the *lowpoint*.

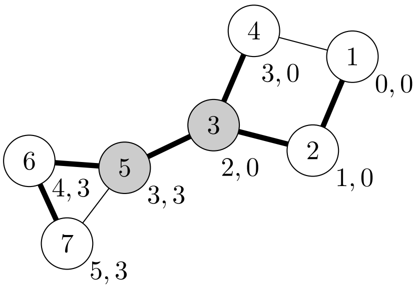


Figure 6.3.: Annotated graph after the depth-first search starting at node 1. The process to determine the annotations is shown in Appendix B.4. Next to each node the depth and lowpoint are noted. The tree traversed by the depth-first search is marked with thick edges. Node 3 is an articulation point, since for a child node the criterion is fulfilled: lowpoint(5) $\geq$ depth(3). 5 is also an articulation point since lowpoint(6) $\geq$ depth(5). Node 1 is not an articulation point, because it has only one child in the tree, i.e., only one *thick* edge. The two biconnected components separated by the articulation nodes are thus constituted by the nodes $\{1, 2, 3, 4\}$ and $\{5, 6, 7\}$.

The depth is basically the number of elements on the stack at the visit of a node and the lowpoint is the smallest depth of any node on the stack which is connected by a path of non-stacked points. As soon as these values for every node are determined, we can identify the *articulation points*, which separate the biconnected components from each other. Namely, u is an articulation point, if the depth of a node u is less or equal than the lowpoint of one of its children in the tree traversed by the depth-first search (marked thick in Figure 6.3), i.e., the nodes visited after u during the depth-first search. For the starting node there is an exception: it is an articulation point iff it has more than one child. Since all annotations are possible in constant time, this algorithm has the same time complexity as the depth-first search of $\mathcal{O}(|V| + |E|)$ [82]. A more detailed and illustrated explanation is available in Appendix B.4.

6.4.2. Erdős-Rényi Graphs

As mentioned above, the Erdős-Rényi graph is probably the first rigorously studied random graph ensemble [141]. It comes in two variants. One with a fixed number of edges $|E| = M$ and one where every edge exists with probability p and has thus a fluctuating number of edges. Here, we will only study the latter type.[6] The *connectivity* of an Erdős-Rényi graph $c = Np$ is equal to the expected degree of of each realization.

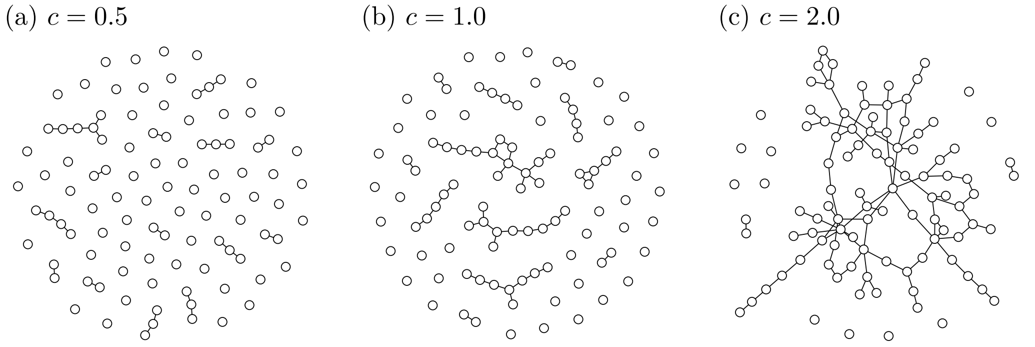

(a) $c = 0.5$ (b) $c = 1.0$ (c) $c = 2.0$

Figure 6.4.: Visualization of Erdős-Rényi graphs for different values of the connectivity c, which determines the average degree $\langle k \rangle = c$ of the resulting graph.

This ensemble is quite well understood, its degree distribution is known to be

$$p_k = \binom{N-1}{k} p^k (1-p)^{N-1-k}, \tag{6.4}$$

the rate function of the size of the largest connected component is known [165] and a percolation transition happens at $c_c = 1$, below which the graph consists of many isolated nodes or tree-like subgraphs with size $\mathcal{O}(1)$ (cf. Figure 6.4(a)) and above which a giant component of size $\mathcal{O}(N)$ exists (cf. Figure 6.4(c)).

The construction of a realization is rather straight forward, one can either start with a graph of N nodes and an empty edge set, iterate over the $N(N-1)$ possible edges and add each with probability $p = c/N$. A faster method would be to first draw the number of edges in the graph M from a binomial distribution and then draw M times two nodes which are not already neighbors and insert an edge between them.

As a specialized Monte Carlo change move to generate Markov chains of Erdős-Rényi graphs (cf. Section 2.2.3), choose a node u uniformly at random, delete all its neighbors and add each possible edge $e \in \{\{u, v\} \mid v \in V \setminus u\}$ with probability p.

6.4.3. Barabási-Albert Graphs

As already mentioned above, Barabási-Albert graph realizations are constructed with a growth process. Since we use a slightly modified approach, we will not describe the

---

[6] But note that Figure 2.4 shows a large part of the distribution for the former case, which is qualitatively very similar to the distribution of the latter case shown in Article A.6.

original model [144], but our modified version.

An ensemble of Barabási-Albert graphs has, apart from the number of nodes N, two parameters: m, the number of edges added for every new node (and therefore $\langle k\rangle = 2m$) and $m_0 \geq m$, the number of nodes with which to start. Independent of the choice of these parameters the degree distribution will follow a power law $p(k) \propto k^{-3}$ for large graphs. While originally only integers were allowed for m, our modification allows also fractional values. This is mainly because by construction for $m = 1$ a tree will arise as in Figure 6.5(a) and for $m = 2$ the whole graph will be a biconnected component as in Figure 6.1(c). Therefore the interesting graphs will arise for values of m in between as shown in Figures 6.5(b) and 6.5(c).

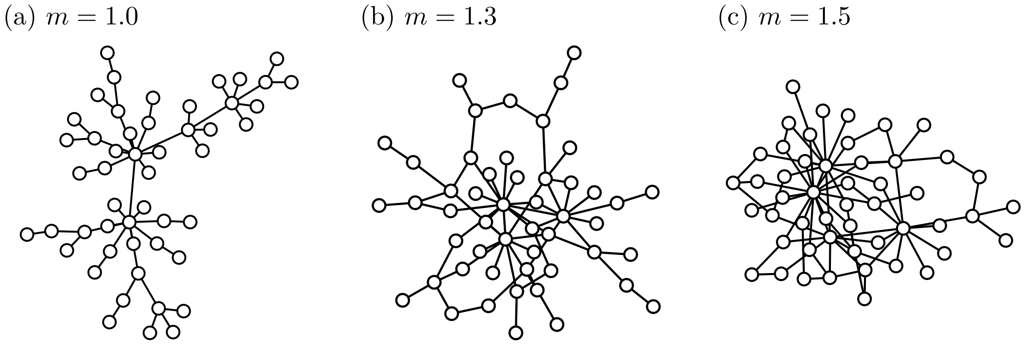

(a) $m = 1.0$ (b) $m = 1.3$ (c) $m = 1.5$

Figure 6.5.: Visualization of Barabási-Albert graphs for different values of m. The case of $m = 1$ in (a) leads to a tree due to the regular construction.

For the construction one starts with a fully connected graph of m_0 nodes. In every iteration one node is added and connected to m' of the already existing nodes. Here our modification comes into play, as a fractional value of m is interpreted as

$$m' = \begin{cases} \lceil m \rceil & \text{with probability } p = m - \lfloor m \rfloor \\ \lfloor m \rfloor & \text{with probability } 1 - p. \end{cases} \tag{6.5}$$

The probability p_i with which an existing node i is connected to the new node is proportional to its degree $p_i = k_i / \sum_j k_j$. This way new nodes will preferentially attach to nodes with a high degree, which will in turn become even more preferable for new nodes.

The usual implementation is to maintain an array where each node i appears k_i times and draw one entry uniformly at random from it to determine a neighbor of a new node. This growth process is iterated until the size of the graph is N. As a change move for Markov chain Monte Carlo simulations the black box approach (cf. Section 2.2.3) can be applied directly and works well.

6.5. Results

Article A.6 finds the distribution of the size of the largest biconnected component S_2 over almost its full support for multiple system sizes and graph ensembles. From this the empirical rate function (cf. Section 2.1) is calculated, which shows even for finite sizes, which were simulated, a convincing convergence, as shown in Figure 6.6(a). Therefore the large deviation principle seems to hold and biconnected components, which do not have the typical size are exponentially suppressed in all scrutinized ensembles.

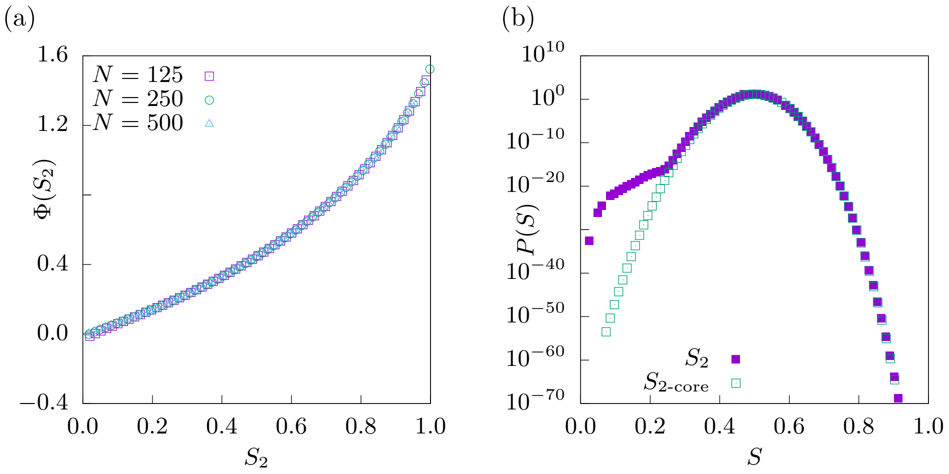


Figure 6.6.: (a) Comparison of the rate function of the size of the largest biconnected component S_2 for Erdős-Rényi graphs of different sizes. The convergence to a limiting form for large sizes is already well visible at these finite sizes. (b) Comparison of the distribution of the size of the largest biconnected components and 2-cores in Barabási-Albert graphs with $N = 500$ nodes and $m = 1.3$. Apparently the behavior is very similar in the main region but qualitatively different in the left tail below probabilities of $P \lesssim 10^{-20}$. (Similar plots are also shown in Article A.6.)

Regarding the comparison with the size of the 2-core, for Erdős-Rényi graphs the behavior is generally very similar to the already known rate function of the 2-core [37] at the same connectivity. The only minor deviation from the 2-core behavior is at very low probabilities in the left tail above the percolation threshold. For Barabási-Albert graphs the results are more interesting. While they are again very similar in the main region, we can observe a qualitative difference in the shape of the distribution in the left tail, which is well visible in Figure 6.6(b). The biconnected component shows a very similar non-monotonous shape as for the Erdős-Rényi graph above the critical connectivity (which coincides with a phase transition in our artifical temperature ensemble,[7] as already shown for the size of the connected component [31]). The size of

---

[7] This can also be observed in Figure 2.4, where the biased histograms $P_\Theta(S)$ at one temperature show a two-peak structure, which is a telltale sign for first order phase transitions.

the 2-core behaves very differently, as the shape of its distribution is convex. Since the deviation only occurs for probabilities of around 10^{-20} or lower, our large deviation approach is necessary to observe this behavior.

Bibliography

[1] D.P. Landau and K. Binder. *A Guide to Monte Carlo Simulations in Statistical Physics*. Cambridge University Press, 2014. ISBN: 9781316062630.

[2] Hugo Touchette. “The large deviation approach to statistical mechanics”. In: *Physics Reports* 478.1–3 (2009), pp. 1–69. ISSN: 0370-1573. DOI: `10.1016/j.physrep.2009.05.002`.

[3] Fabio Cecconi, Massimo Cencini, Andrea Puglisi, Davide Vergni, and Angelo Vulpiani. “From the Law of Large Numbers to Large Deviation Theory in Statistical Physics: An Introduction”. In: *Large Deviations in Physics: The Legacy of the Law of Large Numbers*. Ed. by Angelo Vulpiani, Fabio Cecconi, Massimo Cencini, Andrea Puglisi, and Davide Vergni. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 1–27. ISBN: 978-3-642-54251-0. DOI: `10.1007/978-3-642-54251-0_1`.

[4] Harald Cramér. *Sur un nouveau théorème-limite de la théorie des probabilités*. French. translation at `https://arxiv.org/abs/1802.05988`. 1938.

[5] Marcel Kahlen, Andreas Engel, and Christian Van den Broeck. “Large deviations in Taylor dispersion”. In: *Phys. Rev. E* 95 (1 Jan. 2017), p. 012144. DOI: `10.1103/PhysRevE.95.012144`.

[6] “Dispersion of soluble matter in solvent flowing slowly through a tube”. In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 219.1137 (1953), pp. 186–203. ISSN: 0080-4630. DOI: `10.1098/rspa.1953.0139`.

[7] Marco Baiesi, Alberto Rosso, and Thomas Speck. “Fundamental Problems in Statistical Physics XIV Special Issue”. In: *Physica A: Statistical Mechanics and its Applications* 504 (2018). Lecture Notes of the 14th International Summer School on Fundamental Problems in Statistical Physics, pp. 1–4. ISSN: 0378-4371. DOI: `10.1016/j.physa.2017.10.013`.

[8] Wlodzimierz Bryc. “A remark on the connection between the large deviation principle and the central limit theorem”. In: *Statistics & Probability Letters* 18.4 (1993), pp. 253–256.

[9] Roger Eckhardt. “Stan Ulam, John von Neumann, and the Monte Carlo method”. In: *Los Alamos Science* 15.131-136 (1987), p. 30.

[10] N. Metropolis. “Monte Carlo Method”. In: *From Cardinals to Chaos: Reflection on the Life and Legacy of Stanislaw Ulam* (1989), p. 125.

[11] Herman Kahn. “Use of different Monte Carlo sampling techniques”. In: *RAND Corporation paper series* (1955).

[12] B. Efron. “Bootstrap Methods: Another Look at the Jackknife”. In: *Ann. Statist.* 7.1 (Jan. 1979), pp. 1–26. DOI: `10.1214/aos/1176344552`.

[13] A. Peter Young. *Everything You Wanted to Know About Data Analysis and Fitting but Were Afraid to Ask.* SpringerBriefs in Physics. Springer International Publishing, 2015. ISBN: 978-3-319-19050-1. DOI: `10.1007/978-3-319-19051-8_2`.

[14] Alexander K. Hartmann. *Big Practical Guide to Computer Simulations.* World Scientific, 2015. DOI: `10.1142/9019`.

[15] Peter Grassberger. “Pruned-enriched Rosenbluth method: Simulations of θ polymers of chain length up to 1 000 000”. In: *Phys. Rev. E* 56 (3 Sept. 1997), pp. 3682–3693. DOI: `10.1103/PhysRevE.56.3682`.

[16] Marshall N. Rosenbluth and Arianna W. Rosenbluth. “Monte Carlo calculation of the average extension of molecular chains”. In: *The Journal of Chemical Physics* 23.2 (1955), pp. 356–359.

[17] F. T. Wall and J. J. Erpenbeck. “New Method for the Statistical Computation of Polymer Dimensions”. In: *The Journal of Chemical Physics* 30.3 (1959), pp. 634–637. DOI: `10.1063/1.1730021`.

[18] Ernst Ising. “Beitrag zur Theorie des Ferromagnetismus”. In: *Zeitschrift für Physik* 31.1 (1925), pp. 253–258.

[19] M.E.J. Newman and G.T. Barkema. *Monte Carlo Methods in Statistical Physics.* Clarendon Press, 1999. ISBN: 9780198517979.

[20] Hendrik Schawe, Christoph Norrenbrock, and Alexander K. Hartmann. “Ising Ferromagnets on Proximity Graphs with Varying Disorder of the Node Placement”. In: *Scientific Reports* 7.1 (2017), p. 8040. DOI: `10.1038/s41598-017-08531-8`.

[21] Christoph Dellago, Peter G. Bolhuis, Félix S. Csajka, and David Chandler. “Transition path sampling and the calculation of rate constants”. In: *The Journal of Chemical Physics* 108.5 (1998), pp. 1964–1977.

[22] Hendrik Schawe. *Ising-Ferromagnet auf Ad-Hoc Netzwerken.* 2013.

[23] Peter G. Bolhuis, David Chandler, Christoph Dellago, and Phillip L. Geissler. “Transition path sampling: Throwing ropes over rough mountain passes, in the dark”. In: *Annual review of physical chemistry* 53.1 (2002), pp. 291–318.

[24] Neal Madras and Gordon Slade. “The Self-Avoiding Walk”. In: Springer New York, 2013. Chap. Analysis of Monte Carlo methods, pp. 281–364. ISBN: 978-1-4614-6025-1. DOI: `10.1007/978-1-4614-6025-1_9`.

[25] Manon Michel, Sebastian C. Kapfer, and Werner Krauth. "Generalized event-chain Monte Carlo: Constructing rejection-free global-balance algorithms from infinitesimal steps". In: *The Journal of Chemical Physics* 140.5 (2014), p. 054116. DOI: `10.1063/1.4863991`.

[26] Ulli Wolff. "Collective Monte Carlo Updating for Spin Systems". In: *Phys. Rev. Lett.* 62 (4 Jan. 1989), pp. 361–364. DOI: `10.1103/PhysRevLett.62.361`.

[27] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. "Equation of state calculations by fast computing machines". In: *The Journal of Chemical Physics* 21.6 (1953), pp. 1087–1092.

[28] W. K. Hastings. "Monte Carlo sampling methods using Markov chains and their applications". In: *Biometrika* 57.1 (1970), pp. 97–109. DOI: `10.1093/biomet/57.1.97`.

[29] Alexander K. Hartmann. "High-precision work distributions for extreme nonequilibrium processes in large systems". In: *Phys. Rev. E* 89 (5 May 2014), p. 052103. DOI: `10.1103/PhysRevE.89.052103`.

[30] Makoto Matsumoto and Takuji Nishimura. "Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator". In: *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 8.1 (1998), pp. 3–30.

[31] Alexander K. Hartmann. "Large-deviation properties of largest component for random graphs". In: *The European Physical Journal B* 84.4 (Dec. 2011), pp. 627–634. ISSN: 1434-6036. DOI: `10.1140/epjb/e2011-10836-4`.

[32] Stefan Wolfsheimer, Bernd Burghardt, and Alexander K. Hartmann. "Local sequence alignments statistics: deviations from Gumbel statistics in the rare-event tail". In: *Algorithms for Molecular Biology* 2.1 (2007), p. 9. ISSN: 1748-7188. DOI: `10.1186/1748-7188-2-9`.

[33] Pascal Fieth and Alexander K. Hartmann. "Score distributions of gapped multiple sequence alignments down to the low-probability tail". In: *Phys. Rev. E* 94 (2 Aug. 2016), p. 022127. DOI: `10.1103/PhysRevE.94.022127`.

[34] Matthias Werner, Pascal Fieth, and Alexander K. Hartmann. "Large-Deviation Properties of Sequence Alignment of Correlated Sequences". In: *Journal of Computational Biology* 25.12 (2018), pp. 1339–1346. DOI: `10.1089/cmb.2017.0269`.

[35] Alexander K. Hartmann. "Sampling rare events: Statistics of local sequence alignments". In: *Phys. Rev. E* 65 (5 Apr. 2002), p. 056102. DOI: `10.1103/PhysRevE.65.056102`.

[36] Andreas Engel, Rémi Monasson, and Alexander K. Hartmann. "On Large Deviation Properties of Erdős–Rényi Random Graphs". In: *Journal of Statistical Physics* 117.3 (2004), pp. 387–426. ISSN: 1572-9613. DOI: `10.1007/s10955-004-2268-6`.

[37] Alexander K. Hartmann. “Large-deviation properties of the largest 2-core component for random graphs”. In: *The European Physical Journal Special Topics* 226.4 (Apr. 2017), pp. 567–579. ISSN: 1951-6401. DOI: `10.1140/epjst/e2016-60368-3`.

[38] A. K. Hartmann and M. Mézard. “Distribution of diameters for Erdős-Rényi random graphs”. In: *Phys. Rev. E* 97 (3 Mar. 2018), p. 032128. DOI: `10.1103/PhysRevE.97.032128`.

[39] Gunnar Claussen, Alexander K. Hartmann, and Satya N. Majumdar. “Convex hulls of random walks: Large-deviation properties”. In: *Phys. Rev. E* 91 (5 May 2015), p. 052104. DOI: `10.1103/PhysRevE.91.052104`.

[40] Timo Dewenter, Gunnar Claussen, Alexander K. Hartmann, and Satya N. Majumdar. “Convex hulls of multiple random walks: A large-deviation study”. In: *Phys. Rev. E* 94 (5 Nov. 2016), p. 052120. DOI: `10.1103/PhysRevE.94.052120`.

[41] Jooyoung Lee. “New Monte Carlo algorithm: Entropic sampling”. In: *Phys. Rev. Lett.* 71 (2 July 1993), pp. 211–214. DOI: `10.1103/PhysRevLett.71.211`.

[42] Fugao Wang and D. P. Landau. “Efficient, Multiple-Range Random Walk Algorithm to Calculate the Density of States”. In: *Phys. Rev. Lett.* 86 (10 Mar. 2001), pp. 2050–2053. DOI: `10.1103/PhysRevLett.86.2050`.

[43] Fugao Wang and David P. Landau. “Determining the density of states for classical statistical models: A random walk algorithm to produce a flat histogram”. In: *Phys. Rev. E* 64 (5 Oct. 2001), p. 056101. DOI: `10.1103/PhysRevE.64.056101`.

[44] B. J. Schulz, K. Binder, M. Müller, and D. P. Landau. “Avoiding boundary effects in Wang-Landau sampling”. In: *Phys. Rev. E* 67 (6 June 2003), p. 067102. DOI: `10.1103/PhysRevE.67.067102`.

[45] Thomas Vogel, Ying Wai Li, Thomas Wüst, and David P. Landau. “Generic, Hierarchical Framework for Massively Parallel Wang-Landau Sampling”. In: *Phys. Rev. Lett.* 110 (21 May 2013), p. 210603. DOI: `10.1103/PhysRevLett.110.210603`.

[46] R. E. Belardinelli and V. D. Pereyra. “Fast algorithm to calculate density of states”. In: *Phys. Rev. E* 75 (4 Apr. 2007), p. 046701. DOI: `10.1103/PhysRevE.75.046701`.

[47] R. E. Belardinelli and V. D. Pereyra. “Wang-Landau algorithm: A theoretical analysis of the saturation of the error”. In: *The Journal of Chemical Physics* 127.18, 184105 (2007). DOI: `10.1063/1.2803061`.

[48] Ronald Dickman and A. G. Cunha-Netto. “Complete high-precision entropic sampling”. In: *Phys. Rev. E* 84 (2 Aug. 2011), p. 026701. DOI: `10.1103/PhysRevE.84.026701`.

[49] Karl Pearson. “The problem of the random walk”. In: *Nature* 72.1865 (1905), p. 294. DOI: `10.1038/072294b0`.

[50] Karl Pearson. "The problem of the random walk". In: *Nature* 72.1867 (1905), p. 342. DOI: `10.1038/072342a0`.

[51] Karl Pearson. "A Mathematical Theory of Random Migration, Mathematical Contributions to the Theory of Evolution XV". In: *Draper's Company Research Memoirs, Biometric Series. Dulau and Co, London* (1906).

[52] Georg Pólya. "Über eine Aufgabe der Wahrscheinlichkeitsrechnung betreffend die Irrfahrt im Straßennetz". In: *Mathematische Annalen* 84.1 (1921), pp. 149–160. ISSN: 1432-1807. DOI: `10.1007/BF01458701`.

[53] C. O. Mohr. "Table of Equivalent Populations of North American Small Mammals". In: *American Midland Naturalist* 37.1 (1947), pp. 223–249.

[54] B. J. Worton. "A review of models of home range for animal movement". In: *Ecol. Model.* 38 (1987), pp. 277–298. ISSN: 0304-3800. DOI: `10.1016/0304-3800(87)90101-3`.

[55] S. A. Boyle, W. C. Lourenco, L. R. da Silva, and A. T. Smith. "Home Range Estimates Vary with Sample Size and Methods". In: *Folia Primatol.* 80 (2009), pp. 33–42.

[56] Gérard Letac and Lajos Takács. "Expected Perimeter Length". In: *The American Mathematical Monthly* 87.2 (1980), pp. 142–142. ISSN: 00029890, 19300972. DOI: `10.2307/2322010`.

[57] Gérard Letac. "An explicit calculation of the mean of the perimeter of the convex hull of a plane random walk". In: *Journal of Theoretical Probability* 6.2 (1993), pp. 385–387. ISSN: 1572-9230. DOI: `10.1007/BF01047580`.

[58] Julien Randon-Furling, Satya N. Majumdar, and Alain Comtet. "Convex Hull of N Planar Brownian Motions: Exact Results and an Application to Ecology". In: *Phys. Rev. Lett.* 103 (14 Sept. 2009), p. 140602. DOI: `10.1103/PhysRevLett.103.140602`.

[59] Satya N. Majumdar, Alain Comtet, and Julien Randon-Furling. "Random Convex Hulls and Extreme Value Statistics". In: *Journal of Statistical Physics* 138.6 (2010), pp. 955–1009. ISSN: 1572-9613. DOI: `10.1007/s10955-009-9905-z`.

[60] Ronen Eldan. "Volumetric properties of the convex hull of an n-dimensional Brownian motion". In: *Electron. J. Probab.* 19 (2014), no. 45, 1–34. ISSN: 1083-6489. DOI: `10.1214/EJP.v19-2571`.

[61] Zakhar Kabluchko and Dmitry Zaporozhets. "Intrinsic volumes of Sobolev balls with applications to Brownian convex hulls". In: *Transactions of the American Mathematical Society* 368.12 (2016), pp. 8873–8899. DOI: `10.1090/tran/6628`.

[62] André Goldman. "The spectrum of certain planar Poissonian mosaics and the convex hull of the Brownian bridge". French. In: *J. Prob. Theor. Relat. Fields* 105.1 (1996), pp. 57–83. ISSN: 0178-8051 (print), 1432-2064 (electronic). DOI: `10.1007/BF01192071`.

[63] Timothy Law Snyder and J. Michael Steele. "Convex hulls of random walks". In: *Proceedings of the American Mathematical Society* 117.4 (1993), pp. 1165–1173. DOI: `10.1090/S0002-9939-1993-1169048-2`.

[64] Glen Baxter. "A Combinatorial Lemma for Complex Numbers". In: *Ann. Math. Statist.* 32.3 (Sept. 1961), pp. 901–904. DOI: `10.1214/aoms/1177704985`.

[65] Eric Dumonteil, Satya N. Majumdar, Alberto Rosso, and Andrea Zoia. "Spatial extent of an outbreak in animal epidemics". In: *Proceedings of the National Academy of Sciences* 110.11 (2013), pp. 4239–4244. DOI: `10.1073/pnas.1213237110`.

[66] Ingemar Nåsell. "Stochastic models of some endemic infections". In: *Mathematical Biosciences* 179.1 (2002), pp. 1–19. ISSN: 0025-5564. DOI: `10.1016/S0025-5564(02)00098-6`.

[67] Cristopher Moore and M. E. J. Newman. "Epidemics and percolation in small-world networks". In: *Phys. Rev. E* 61 (5 May 2000), pp. 5678–5682. DOI: `10.1103/PhysRevE.61.5678`.

[68] Romualdo Pastor-Satorras and Alessandro Vespignani. "Epidemic Spreading in Scale-Free Networks". In: *Phys. Rev. Lett.* 86 (14 Apr. 2001), pp. 3200–3203. DOI: `10.1103/PhysRevLett.86.3200`.

[69] Yann Lanoiselée and Denis S. Grebenkov. "Unraveling intermittent features in single-particle trajectories by a local convex hull method". In: *Phys. Rev. E* 96 (2 Aug. 2017), p. 022144. DOI: `10.1103/PhysRevE.96.022144`.

[70] Howard C. Berg and Douglas A Brown. "Chemotaxis in Escherichia coli analysed by three-dimensional tracking". In: *nature* 239.5374 (1972), pp. 500–504. DOI: `10.1038/239500a0`.

[71] Arseniy Akopyan and Vladislav Vysotsky. "Large deviations for the perimeter of convex hulls of planar random walks". In: *arXiv preprint arXiv:1606.07141* (2016).

[72] A.L. Cauchy. *Mémoire sur la rectification des courbes et la quadrature des surfaces courbées.* French. 1832.

[73] Neal Madras and Alan D. Sokal. "The pivot algorithm: A highly efficient Monte Carlo method for the self-avoiding walk". In: *Journal of Statistical Physics* 50.1 (1988), pp. 109–186. ISSN: 1572-9613. DOI: `10.1007/BF01022990`.

[74] Keizo Suzuki. "The Excluded Volume Effect of Very-long-chain Molecules". In: *Bulletin of the Chemical Society of Japan* 41.2 (1968), pp. 538–538. DOI: `10.1246/bcsj.41.538`.

[75] Nathan Clisby. "Accurate Estimate of the Critical Exponent ν for Self-Avoiding Walks via a Fast Implementation of the Pivot Algorithm". In: *Phys. Rev. Lett.* 104 (5 Feb. 2010), p. 055702. DOI: `10.1103/PhysRevLett.104.055702`.

[76] Nathan Clisby. “Efficient Implementation of the Pivot Algorithm for Self-avoiding Walks”. In: *Journal of Statistical Physics* 140.2 (July 2010), pp. 349–392. ISSN: 1572-9613. DOI: `10.1007/s10955-010-9994-8`.

[77] Gregory F. Lawler. “A self-avoiding random walk.” In: *Duke Math. J.* 47 (1980), pp. 655–693. ISSN: 0012-7094; 1547-7398/e. DOI: `10.1215/S0012-7094-80-04741-9`.

[78] Gregory F. Lawler. “Loop-Erased Random Walk”. In: *Perplexing Problems in Probability: Festschrift in Honor of Harry Kesten.* Ed. by Maury Bramson and Rick Durrett. Boston, MA: Birkhäuser Boston, 1999, pp. 197–217. ISBN: 978-1-4612-2168-5. DOI: `10.1007/978-1-4612-2168-5_12`.

[79] S. N. Majumdar. “Exact fractal dimension of the loop-erased self-avoiding walk in two dimensions”. In: *Phys. Rev. Lett.* 68 (15 Apr. 1992), pp. 2329–2331. DOI: `10.1103/PhysRevLett.68.2329`.

[80] Abel Weinrib and S. A. Trugman. “A new kinetic walk and percolation perimeters”. In: *Phys. Rev. B* 31 (5 Mar. 1985), pp. 2993–2997. DOI: `10.1103/PhysRevB.31.2993`.

[81] K. Kremer and J. W. Lyklema. “Indefinitely Growing Self-Avoiding Walk”. In: *Phys. Rev. Lett.* 54 (4 Jan. 1985), pp. 267–269. DOI: `10.1103/PhysRevLett.54.267`.

[82] Thomas H. Cormen, Charles Eric Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to algorithms.* MIT press Cambridge, 2009.

[83] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. “A formal basis for the heuristic determination of minimum cost paths”. In: *IEEE transactions on Systems Science and Cybernetics* 4.2 (1968), pp. 100–107.

[84] Daniel J. Amit, G. Parisi, and L. Peliti. “Asymptotic behavior of the "true" self-avoiding walk”. In: *Phys. Rev. B* 27 (3 Feb. 1983), pp. 1635–1645. DOI: `10.1103/PhysRevB.27.1635`.

[85] L. Pietronero. “Critical dimensionality and exponent of the "true" self-avoiding walk”. In: *Phys. Rev. B* 27 (9 May 1983), pp. 5887–5889. DOI: `10.1103/PhysRevB.27.5887`.

[86] Barry D Hughes. *Random walks and random environments.* Clarendon Press Oxford, 1996. ISBN: 9780198537885.

[87] Neal Madras and Gordon Slade. “The Self-Avoiding Walk”. In: New York, NY: Springer New York, 2013. Chap. Introduction, pp. 281–364. ISBN: 978-1-4614-6025-1. DOI: `10.1007/978-1-4614-6025-1_9`.

[88] David B. Wilson. “Dimension of the loop-erased random walk in three dimensions”. In: *Phys. Rev. E* 82 (6 Dec. 2010), p. 062102. DOI: `10.1103/PhysRevE.82.062102`.

[89] Franco P. Preparata and Michael I. Shamos. *Computational Geometry: An Introduction.* Springer Science & Business Media, 2012. ISBN: 978-0387961316.

[90] Franz Aurenhammer. “Voronoi Diagrams – a Survey of a Fundamental Geometric Data Structure”. In: *ACM Comput. Surv.* 23.3 (Sept. 1991), pp. 345–405. ISSN: 0360-0300. DOI: `10.1145/116873.116880`.

[91] Victor Klee. “On the complexity of d-dimensional Voronoi diagrams”. In: *Archiv der Mathematik* 34.1 (1980), pp. 75–80. DOI: `10.1007/BF01224932`.

[92] Raimund Seidel. “A convex hull algorithm optimal for point sets in even dimensions”. PhD thesis. University of British Columbia, 1981. DOI: `10.14288/1.0051821`.

[93] Kenneth L. Clarkson and Peter W. Shor. “Applications of random sampling in computational geometry, II”. In: *Discrete & Computational Geometry* 4.5 (1989), pp. 387–421. ISSN: 1432-0444. DOI: `10.1007/BF02187740`.

[94] Zong-Ben Xu, Jiang-She Zhang, and Yiu-Wing Leung. “An approximate algorithm for computing multidimensional convex hulls”. In: *Applied Mathematics and Computation* 94.2–3 (1998), pp. 193–226. ISSN: 0096-3003. DOI: `10.1016/S0096-3003(97)10043-1`.

[95] Hossein Sartipizadeh and Tyrone L Vincent. “Computing the approximate convex hull in high dimensions”. In: *arXiv preprint arXiv:1603.04422* (2016).

[96] A.M. Andrew. “Another efficient algorithm for convex hulls in two dimensions”. In: *Information Processing Letters* 9.5 (1979), pp. 216–219. ISSN: 0020-0190. DOI: `10.1016/0020-0190(79)90072-3`.

[97] Donald Ervin Knuth. *The Art of Computer Programming: Sorting and Searching*. Vol. 3. Addison-Wesley, 1997. ISBN: 0-201-89685-0.

[98] Ronald L. Graham. “An Efficient Algorithm for Determining the Convex Hull of a Finite Planar Set”. In: *Information Processing Letters* 1 (1972), pp. 132–133. DOI: `10.1016/0020-0190(72)90045-2`.

[99] Kenneth R. Anderson. “A reevaluation of an efficient algorithm for determining the convex hull of a finite planar set”. In: *Information Processing Letters* 7.1 (1978), pp. 53–55. ISSN: 0020-0190. DOI: `10.1016/0020-0190(78)90041-8`.

[100] William F. Eddy. “A New Convex Hull Algorithm for Planar Sets”. In: *ACM Trans. Math. Softw.* 3.4 (Dec. 1977), pp. 398–403. ISSN: 0098-3500. DOI: `10.1145/355759.355766`.

[101] A. Bykat. “Convex hull of a finite set of points in two dimensions”. In: *Information Processing Letters* 7.6 (1978), pp. 296–298. ISSN: 0020-0190. DOI: `10.1016/0020-0190(78)90021-2`.

[102] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. “The Quickhull algorithm for convex hulls”. In: *ACM Trans. Math. Softw.* 22.4 (1996), pp. 469–483. DOI: `10.1.1.117.405`.

[103] Ernst Mücke. “Quickhull: Computing Convex Hulls Quickly”. In: *Computing in Science & Engineering* 11.5 (2009), pp. 54–57. DOI: `10.1109/MCSE.2009.136`.

[104] Imre Bárány and Zoltán Füredi. “Computing the volume is difficult”. In: *Discrete & Computational Geometry* 2.4 (1987), pp. 319–326. ISSN: 1432-0444. DOI: `10.1007/BF02187886`.

[105] Benno Büeler, Andreas Enge, and Komei Fukuda. “Exact Volume Computation for Polytopes: A Practical Study”. In: *Polytopes — Combinatorics and Computation*. Ed. by Gil Kalai and Günter M. Ziegler. Basel: Birkhäuser Basel, 2000, pp. 131–154. ISBN: 978-3-0348-8438-9. DOI: `10.1007/978-3-0348-8438-9_6`.

[106] P. Stein. “A Note on the Volume of a Simplex”. In: *The American Mathematical Monthly* 73.3 (1966), pp. 299–301. ISSN: 00029890, 19300972. DOI: `10.2307/2315353`.

[107] H. N. Nagaraja. “Order Statistics from Independent Exponential Random Variables and the Sum of the Top Order Statistics”. In: *Advances in Distribution Theory, Order Statistics, and Inference*. Ed. by N. Balakrishnan, José María Sarabia, and Enrique Castillo. Boston, MA: Birkhäuser Boston, 2006, pp. 173–185. ISBN: 978-0-8176-4487-1. DOI: `10.1007/0-8176-4487-3_11`.

[108] Bernard Derrida. “Random-Energy Model: Limit of a Family of Disordered Models”. In: *Phys. Rev. Lett.* 45 (2 July 1980), pp. 79–82. DOI: `10.1103/PhysRevLett.45.79`.

[109] Bernard Derrida. “Random-energy model: An exactly solvable model of disordered systems”. In: *Phys. Rev. B* 24 (5 Sept. 1981), pp. 2613–2626. DOI: `10.1103/PhysRevB.24.2613`.

[110] David Sherrington and Scott Kirkpatrick. “Solvable Model of a Spin-Glass”. In: *Phys. Rev. Lett.* 35 (26 Dec. 1975), pp. 1792–1796. DOI: `10.1103/PhysRevLett.35.1792`.

[111] A. Peter Young. *Spin glasses and random fields*. Vol. 12. World Scientific, 1998. ISBN: 978-981-02-3240-5. DOI: `10.1142/3517`.

[112] Marc Mézard, Giorgio Parisi, and Miguel Virasoro. *Spin glass theory and beyond: An Introduction to the Replica Method and Its Applications*. Vol. 9. World Scientific Publishing Company, 1987.

[113] Giorgio Parisi. “Infinite Number of Order Parameters for Spin-Glasses”. In: *Phys. Rev. Lett.* 43 (23 Dec. 1979), pp. 1754–1756. DOI: `10.1103/PhysRevLett.43.1754`.

[114] Giorgio Parisi. “Order Parameter for Spin-Glasses”. In: *Phys. Rev. Lett.* 50 (24 June 1983), pp. 1946–1948. DOI: `10.1103/PhysRevLett.50.1946`.

[115] David J. Gross and Marc Mézard. “The simplest spin glass”. In: *Nuclear Physics B* 240.4 (1984), pp. 431–452.

[116] Michel Talagrand. *Spin glasses: a challenge for mathematicians: cavity and mean field models*. Vol. 46. Springer Science & Business Media, 2003.

[117] Frank R. De Hoog, J.H. Knight, and A.N. Stokes. “An improved method for numerical inversion of Laplace transforms”. In: *SIAM Journal on Scientific and Statistical Computing* 3.3 (1982), pp. 357–366.

[118] Fredrik Johansson et al. *mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 1.0.0)*. `http://mpmath.org/`. Dec. 2013.

[119] Stanislaw M. Ulam. *Monte Carlo Calculations in Problems of Mathematical Physics*. Ed. by E.F. Beckenbach and M.R. Hestenes. Dover Books on Engineering Series. Dover Publications, Incorporated, 2013. Chap. 11, pp. 261–281. ISBN: 9780486497471.

[120] Kurt Johansson. “The longest increasing subsequence in a random permutation and a unitary random matrix model”. In: *Mathematical Research Letters* 5.1 (1998), pp. 68–82.

[121] Timo Seppäläinen. “Large deviations for increasing sequences on the plane”. In: *Probability Theory and Related Fields* 112.2 (Oct. 1998), pp. 221–244. ISSN: 1432-2064. DOI: `10.1007/s004400050188`.

[122] Benjamin F. Logan and Larry A. Shepp. “A variational problem for random Young tableaux”. In: *Young Tableaux in Combinatorics, Invariant Theory, and Algebra*. Elsevier, 1977.

[123] Jean-Dominique Deuschel and Ofer Zeitouni. “On increasing subsequences of IID samples”. In: *Combinatorics, Probability and Computing* 8.3 (1999), pp. 247–263. DOI: `10.1017/S0963548399003776`.

[124] Jinho Baik, Percy Deift, and Kurt Johansson. “On the distribution of the length of the longest increasing subsequence of random permutations”. In: *Journal of the American Mathematical Society* 12.4 (1999), pp. 1119–1178.

[125] Craig A. Tracy and Harold Widom. “Level-spacing distributions and the Airy kernel”. In: *Communications in Mathematical Physics* 159.1 (Jan. 1994), pp. 151–174. ISSN: 1432-0916. DOI: `10.1007/BF02100489`.

[126] Kazumasa A. Takeuchi. “An appetizer to modern developments on the Kardar–Parisi–Zhang universality class”. In: *Physica A: Statistical Mechanics and its Applications* 504 (2018). Lecture Notes of the 14th International Summer School on Fundamental Problems in Statistical Physics, pp. 77–105. ISSN: 0378-4371. DOI: `10.1016/j.physa.2018.03.009`.

[127] Michael Prähofer and Herbert Spohn. “Universal Distributions for Growth Processes in $1+1$ Dimensions and Random Matrices”. In: *Phys. Rev. Lett.* 84 (21 May 2000), pp. 4882–4885. DOI: `10.1103/PhysRevLett.84.4882`.

[128] Kurt Johansson. “Shape Fluctuations and Random Matrices”. In: *Communications in Mathematical Physics* 209.2 (Feb. 2000), pp. 437–476. ISSN: 1432-0916. DOI: `10.1007/s002200050027`.

[129] Satya N. Majumdar and Sergei Nechaev. "Anisotropic ballistic deposition model with links to the Ulam problem and the Tracy-Widom distribution". In: *Phys. Rev. E* 69 (1 Jan. 2004), p. 011103. DOI: `10.1103/PhysRevE.69.011103`.

[130] Satya N. Majumdar. "Monte Carlo Calculations in Problems of Mathematical Physics". In: *Complex Systems: Lecture Notes of the Les Houches Summer School 2006*. Ed. by J.P. Bouchaud, M. Mézard, and J. Dalibard. Les Houches. Elsevier Science, 2006. Chap. 4. ISBN: 9780080550596.

[131] Kazumasa A. Takeuchi and Masaki Sano. "Universal Fluctuations of Growing Interfaces: Evidence in Turbulent Liquid Crystals". In: *Phys. Rev. Lett.* 104 (23 June 2010), p. 230601. DOI: `10.1103/PhysRevLett.104.230601`.

[132] Kazumasa A. Takeuchi, Masaki Sano, Tomohiro Sasamoto, and Herbert Spohn. "Growing interfaces uncover universal fluctuations behind scale invariance". In: *Scientific Reports* 1 (2011), p. 34. DOI: `doi.org/10.1038/srep00034`.

[133] J. Ricardo G. Mendonça. "Empirical scaling of the length of the longest increasing subsequences of random walks". In: *Journal of Physics A: Mathematical and Theoretical* 50.8 (2017), 08LT02. DOI: `10.1088/1751-8121/aa56a3`.

[134] Omer Angel, Richard Balka, and Yuval Peres. "Increasing subsequences of random walks". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 163.1 (2017), pp. 173–185. DOI: `10.1017/S0305004116000797`.

[135] Robin Pemantle and Yuval Peres. "Non-universality for longest increasing subsequence of a random walk". In: *Latin American Journal of Probability and Mathematical Statistics* 14 (2017), pp. 327–336. ISSN: 1980-0436.

[136] Achim Klenke. "Unbegrenzt teilbare Verteilungen". In: *Wahrscheinlichkeitstheorie*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 337–356. ISBN: 978-3-642-36018-3. DOI: `10.1007/978-3-642-36018-3_16`.

[137] Donald Ervin Knuth. *The Art of Computer Programming: Seminumerical Algorithms*. Vol. 2. Addison-Wesley, 1998. ISBN: 0-201-89684-2.

[138] C. L. Mallows. "Problem 62-2". In: *SIAM Review* 5.4 (1963), pp. 375–376. ISSN: 00361445.

[139] Badrish Chandramouli and Jonathan Goldstein. "Patience is a Virtue: Revisiting Merge and Sort on Modern Processors". In: *ACM SIGMOD International Conference on Management of Data (SIGMOD 2014)*. ACM SIGMOD, June 2014.

[140] David Aldous and Persi Diaconis. "Longest increasing subsequences: from patience sorting to the Baik-Deift-Johansson theorem". In: *Bulletin of the American Mathematical Society* 36.4 (1999), pp. 413–432. DOI: `10.1090/S0273-0979-99-00796-X`.

[141] Paul Erdős and Alfréd Rényi. "On the evolution of random graphs". In: *Publ. Math. Inst. Hung. Acad. Sci.* 5.1 (1960), pp. 17–60.

[142] S. Redner. "How popular is your paper? An empirical study of the citation distribution". In: *The European Physical Journal B - Condensed Matter and Complex Systems* 4.2 (July 1998), pp. 131–134. ISSN: 1434-6036. DOI: `10.1007/s100510050359`.

[143] Réka Albert, Hawoong Jeong, and Albert-László Barabási. "Internet: Diameter of the world-wide web". In: *nature* 401.6749 (1999), p. 130. DOI: `10.1038/43601`.

[144] Albert-László Barabási and Réka Albert. "Emergence of scaling in random networks". In: *science* 286.5439 (1999), pp. 509–512.

[145] Duncan J. Watts and Steven H. Strogatz. "Collective dynamics of 'small-world' networks". In: *nature* 393.6684 (1998), p. 440. DOI: `10.1038/30918`.

[146] Jeffrey Travers and Stanley Milgram. "The small world problem". In: *Phychology Today* 1.1 (1967), pp. 61–67.

[147] Jure Leskovec and Andrej Krevl. *SNAP Datasets: Stanford Large Network Dataset Collection*. `http://snap.stanford.edu/data`. June 2014.

[148] Ryan A. Rossi and Nesreen K. Ahmed. "The Network Data Repository with Interactive Graph Analytics and Visualization". In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. 2015.

[149] Jérôme Kunegis. "Konect: the koblenz network collection". In: *Proceedings of the 22nd International Conference on World Wide Web*. ACM. 2013, pp. 1343–1350.

[150] Derek De Solla Price. "A general theory of bibliometric and other cumulative advantage processes". In: *Journal of the American Society for Information Science* 27.5 (1976), pp. 292–306. DOI: `10.1002/asi.4630270505`.

[151] Reuven Cohen, Keren Erez, Daniel ben-Avraham, and Shlomo Havlin. "Resilience of the Internet to Random Breakdowns". In: *Phys. Rev. Lett.* 85 (21 Nov. 2000), pp. 4626–4628. DOI: `10.1103/PhysRevLett.85.4626`.

[152] Albert-László Barabási. "Scale-Free Networks: A Decade and Beyond". In: *Science* 325.5939 (2009), pp. 412–413. ISSN: 0036-8075. DOI: `10.1126/science.1173299`.

[153] Réka Albert, Hawoong Jeong, and Albert-László Barabási. "Error and attack tolerance of complex networks". In: *nature* 406.6794 (2000), p. 378. DOI: `10.1038/35019019`.

[154] M. L. Sachtjen, B. A. Carreras, and V. E. Lynch. "Disturbances in a power transmission system". In: *Phys. Rev. E* 61 (5 May 2000), pp. 4877–4882. DOI: `10.1103/PhysRevE.61.4877`.

[155] M. Rohden, Andreas S., M. Timme, and D. Witthaut. "Self-Organized Synchronization in Decentralized Power Grids". In: *Phys. Rev. Lett.* 109 (6 Aug. 2012), p. 064101. DOI: `10.1103/PhysRevLett.109.064101`.

[156] Timo Dewenter and Alexander K. Hartmann. “Large-deviation properties of resilience of power grids”. In: *New Journal of Physics* 17.1 (2015), p. 015005. DOI: `10.1088/1367-2630/17/1/015005`.

[157] D.-S. Lee and H. Rieger. “Maximum flow and topological structure of complex networks”. In: *EPL (Europhysics Letters)* 73.3 (2006), p. 471.

[158] Duncan S. Callaway, M. E. J. Newman, Steven H. Strogatz, and Duncan J. Watts. “Network Robustness and Fragility: Percolation on Random Graphs”. In: *Phys. Rev. Lett.* 85 (25 Dec. 2000), pp. 5468–5471. DOI: `10.1103/PhysRevLett.85.5468`.

[159] M. E. J. Newman and Gourab Ghoshal. “Bicomponents and the Robustness of Networks to Failure”. In: *Phys. Rev. Lett.* 100 (13 Mar. 2008), p. 138701. DOI: `10.1103/PhysRevLett.100.138701`.

[160] C. Norrenbrock, O. Melchert, and A. K. Hartmann. “Fragmentation properties of two-dimensional proximity graphs considering random failures and targeted attacks”. In: *Phys. Rev. E* 94 (6 Dec. 2016), p. 062125. DOI: `10.1103/PhysRevE.94.062125`.

[161] A.L. Barabási. *Network Science*. Cambridge University Press, 2016. Chap. 4. ISBN: 9781107076266.

[162] Gourab Ghoshal. “Structural and Dynamical Properties of Complex Networks.” PhD thesis. University of Michigan, 2009.

[163] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. “Random graphs with arbitrary degree distributions and their applications”. In: *Phys. Rev. E* 64 (2 July 2001), p. 026118. DOI: `10.1103/PhysRevE.64.026118`.

[164] Scott L. Feld. “Why Your Friends Have More Friends Than You Do”. In: *American Journal of Sociology* 96.6 (1991), pp. 1464–1477. DOI: `10.1086/229693`.

[165] Marek Biskup, Lincoln Chayes, and S. Alex Smith. “Large-deviations / thermodynamic approach to percolation on the complete graph”. In: *Random Structures & Algorithms* 31.3 (2007), pp. 354–370.

[166] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes. “k-Core Organization of Complex Networks”. In: *Phys. Rev. Lett.* 96 (4 Feb. 2006), p. 040601. DOI: `10.1103/PhysRevLett.96.040601`.

[167] John Hopcroft and Robert Tarjan. “Algorithm 447: Efficient Algorithms for Graph Manipulation”. In: *Commun. ACM* 16.6 (June 1973), pp. 372–378. ISSN: 0001-0782. DOI: `10.1145/362248.362272`.

[168] Stefan Wolfsheimer. “Entropy functions and rare events in disordered systems by transfer matrix calculations and Monte Carlo sampling”. PhD thesis. Carl von Ossietzky Universität Oldenburg, 2009.

[169] Selim G. Akl and Godfried T. Toussaint. “A fast convex hull algorithm”. In: *Information Processing Letters* 7.5 (1978), pp. 219–222. ISSN: 0020-0190. DOI: `10.1016/0020-0190(78)90003-0`.

[170] Jean Souviron. "Convex hull: Incremental variations on the Akl-Toussaint heuristics Simple, optimal and space-saving convex hull algorithms". In: *arXiv preprint arXiv:1304.2676* (2013).

[171] R.A. Jarvis. "On the identification of the convex hull of a finite set of points in the plane". In: *Information Processing Letters* 2.1 (1973), pp. 18–21. ISSN: 0020-0190. DOI: `10.1016/0020-0190(73)90020-3`.

[172] T. M. Chan. "Optimal output-sensitive convex hull algorithms in two and three dimensions". In: *Discrete & Computational Geometry* 16.4 (1996), pp. 361–368. ISSN: 1432-0444. DOI: `10.1007/BF02712873`.

[173] D. Kirkpatrick and R. Seidel. "The Ultimate Planar Convex Hull Algorithm?" In: *SIAM Journal on Computing* 15.1 (1986), pp. 287–299. DOI: `10.1137/0215021`.

[174] B. Chazelle and D. P. Dobkin. "Intersection of Convex Objects in Two and Three Dimensions". In: *J. ACM* 34.1 (Jan. 1987), pp. 1–27. ISSN: 0004-5411. DOI: `10.1145/7531.24036`.

[175] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[176] G. Marsaglia and T. A. Bray. "A Convenient Method for Generating Normal Variables". In: *SIAM Review* 6.3 (1964), pp. 260–264. ISSN: 00361445.

[177] George Marsaglia and Wai Wan Tsang. "The ziggurat method for generating random variables". In: *Journal of statistical software* 5.8 (2000), pp. 1–7. DOI: `10.18637/jss.v005.i08`.

[178] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth. "On the Lambert$W$ function". In: *Advances in Computational Mathematics* 5.1 (Dec. 1996), pp. 329–359. ISSN: 1572-9044. DOI: `10.1007/BF02124750`.

Index and Glossary

- **A*** A heuristic best-first search algorithm, to find paths on graphs. 43
- **autocorrelation** 17, 22
- **AVL tree** A datastructure representing an ordered set. This is a self-balancing version of a binary tree. The name is derived from the initials of the inventors. 61

- **ballistic deposition** 69
- **black box approach** 18, 36, 38, 42, 44, 46, 62, 86
- **Box-Muller method** 175

- **Cauchy's formula** 33
- **central limit theorem** 6, 57
- **convex hull** 29, 48
  - **Akl-Toussaint heuristic** 170
  - **Andrew's monotone chain** 51, 172
  - **Chan's algorithm** 172
    - **sub-hull** 172, 173
  - **facet** 49
    - **normal** ($\boldsymbol{n}$) 49
  - **hypervolume** (V) 49, 54
  - **Jarvis' march** 171, 172
  - **Quickhull** 52
    - **eye point** 53
    - **horizon** 53
  - **ridge** 49
  - **surface** (∂V) 49, 54
- **crankshaft move** 19, 38
- **cumulant generating function** 7

- **depth-first search** 43, 53, 84, 176–178
- **detailed balance** 16, 22, 23, 25
- **dimerization** 40, 41
- **directed acyclic graph** 74

- **entropic sampling** 23
- **equilibration** 17, 22, 24
- **ergodicity** 15, 24
- **extreme-value theory** 29, 34, 57

- **flat histogram** 23

- **Gaussian orthogonal ensemble** 68, 69
- **Gaussian symplectic ensemble** 68
- **Gaussian unitary ensemble** 68, 69
- **global balance** 16
- **GMP** A very fast, free and open source multiprecision library `https://gmplib.org/`. 64
- **GNU Scientific Library** A free and open source C library to calculate many fundamental functions and related applications in the field of scientific computing `https://www.gnu.org/software/gsl/`. 176
- **Gärtner-Ellis theorem** 6
- **graph** (G) 77, 79
  - **Barabási-Albert graph** 78, 80, 83, 85–87
  - **biconnected component** 80, 81, 83
  - **bi-edge-connected component** 22, 80
  - **bi-node-connected component** *see* biconnected component
  - **connected** 80
  - **connected component** 80
  - **cycle** 80
  - **degree** 80
  - **degree distribution** 80
  - **diameter** 80
  - **edge set** (E) 79
  - **edge-independent** 80

- **Erdős-Rényi graph** 22, 77, 78, 80, 81, 83, 85, 87, 158
- **excess degree distribution** 82
- **incident edge** 80
- **neighbor** 80
- **node set** (V) 79
- **node-independent** 80–82
- **path** 80
- **scale-free** 78, 80
- **small-world** 30, 78
- **subgraph** 80

hash set A datastructure representing a set. The implementation is similar to hash tables and offers amortized constant time insertions, deletions and finding of the unique elements. 41, 42

i.i.d. identically, independently distributed 30, 57, 60, 68, 71
importance sampling 9, 12
inversion method A general technique to generate random numbers according to a distribution, whose cumulative distribution function is invertable. 176
Ising model 13, 59

KPZ Kardar-Parisi-Zhang 68, 69

Laplace transform 6, 63
Laplace's approximation 7
large deviation principle 3, 4, 32, 87
Large deviation theory 3
law of large numbers 6
Legendre transform 6
lexicographical sorting 51
LIS longest increasing subsequence 67–75

Markov chain 15, 23
Markov process 15
Markov property 15
master equation 16
MCMC Markov chain Monte Carlo 15, 17, 36, 40, 41, 61, 86
merge sort An divide-and-conquer sorting algorithm. Sorts subsets of the problem and merges them cleverly. 72
Mersenne Twister 19, 175
Metropolis algorithm 13
- **Metropolis acceptance probability** 17
- **Metropolis-Hastings algorithm** 17

moment generating function 6
mpmath A free and open source multiprecision library offering implementations of many advanced operations `http://mpmath.org/`. 64

patience sort 72
PERM pruned enriched Rosenbluth method 11, 12, 41
phase transition 59, 87
pivot move 40, 41
polynuclear growth 69

qhull A free and open source C/C++ library to calculate convex hulls of point sets in arbitrary dimensions `http://www.qhull.org/`. 174

random walk 29, 35, 36
- **Gaussian random walk** 38, 48
- **lattice random walk** 19, 36, 38, 41–44, 46–48
- **loop-erased random walk** 41, 42, 44, 45, 47, 48, 55
- **self-avoiding random walk** 9, 11, 12, 15, 38, 40–45, 47, 48, 55
- **smart-kinetic self-avoiding random walk** 43–46, 48, 55, 56
- **"true" self-avoiding random walk** 45, 46, 48, 56, 130

random-energy model 58
rate function (Φ) 4, 35, 55, 87
red-black tree A datastructure representing an ordered set. This is a self-balancing version of a binary tree. 61

rejection method A general technique to generate random numbers according to a distribution. 175
replica-exchange Wang Landau 24

scaled cumulant generating function 6
shoelace formula 54
simple sampling 9
SIR model 30, 78
SK Sherrington-Kirkpatrick 58, 59
subsequence 67

Tracy-Widom distribution 68–71
transition path sampling 13, 14
tree 80

Ulam's problem 67, *see* LIS

Wang Landau sampling 23–27
winding angle 43

Ziggurat method A modern procedure to generate a Gaussian random number from uniform random numbers, based on the rejection method. 175

Appendices

A. Publications

This chapter shows the central publications of this thesis. Articles A.1 and A.2 are both published in the peer-reviewed journal *Physical Review E.* Article A.3 is accepted by the *Journal of Physics: Conference Series*, a peer-reviewed, open-access journal. Article A.4 is published in the peer-reviewed journal *Europhysics Letters.* Article A.5 is published in the peer-reviewed journal *Physical Review E.* Article A.6 is published in the peer-reviewed *European Physical Journal B.*

Each of these publications, respectively drafts, will be printed with a concise statement specifying the contributions of all coauthors in Sections A.1 to A.6.

A.1. Convex hulls of random walks in higher dimensions: A large-deviation study 110
A.2. Large deviations of convex hulls of self-avoiding random walks 120
A.3. Large Deviations of Convex Hulls of the “True” Self-Avoiding Random Walk 130
A.4. Ground state energy of noninteracting fermions with a random energy spectrum 142
A.5. Large deviations of the length of the longest increasing subsequence of random permutations and random walks 150
A.6. Large-deviation properties of the largest biconnected component for random graphs 158

The correctness of the statements detailing the contributions is confirmed by the adviser of this thesis.

(Signature: Prof. Dr. Alexander K. Hartmann)

A.1. Convex hulls of random walks in higher dimensions: A large-deviation study

The first author, Hendrik Schawe, is the author of the thesis at hand. Alexander K. Hartmann is the supervising professor of H. Schawe. Satya N. Majumdar is directeur de recherche at the Laboratoire de Physique Théorique et Modèles Statistiques (LPTMS) at the Université Paris–Sud in Orsay, France

This publication is a follow-up project to Reference [39], which looked at a very similar problem in the plane, and Reference [40], which belongs to the same DFG grant HA 3169/8-1 and is also limited to the plane. Since the publication at hand operates in higher dimensions no code was shared from the above mentioned projects and simulation and evaluation programs were written from scratch by H. Schawe. During frequent meetings of H. Schawe with A. K. Hartmann and some meetings with S. N. Majumdar during a one month stay at the LPTMS, the state and target of the project and possible interesting quantities were discussed. The first draft was prepared by H. Schawe with direct feedback from A. K. Hartmann. At this stage S. N. Majumdar gave some ideas for further improvements, which were incorporated.

PHYSICAL REVIEW E **96**, 062101 (2017)

Convex hulls of random walks in higher dimensions: A large-deviation study

Hendrik Schawe[*] and Alexander K. Hartmann[†]
Institut für Physik, Universität Oldenburg, 26111 Oldenburg, Germany and LPTMS, CNRS, Université Paris-Sud, Université Paris-Saclay, 91405 Orsay, France

Satya N. Majumdar[‡]
LPTMS, CNRS, Université Paris-Sud, Université Paris-Saclay, 91405 Orsay, France

(Received 11 September 2017; revised manuscript received 14 November 2017; published 1 December 2017)

The distribution of the hypervolume V and surface ∂V of convex hulls of (multiple) random walks in higher dimensions are determined numerically, especially containing probabilities far smaller than $P = 10^{-1000}$ to estimate large deviation properties. For arbitrary dimensions and large walk lengths T, we suggest a scaling behavior of the distribution with the length of the walk T similar to the two-dimensional case and behavior of the distributions in the tails. We underpin both with numerical data in $d = 3$ and $d = 4$ dimensions. Further, we confirm the analytically known means of those distributions and calculate their variances for large T.

DOI: 10.1103/PhysRevE.96.062101

I. INTRODUCTION

The random walk (RW) is first mentioned [1] with this name in 1905 by Pearson [2] as a model, where at discrete times, steps of a fixed length are taken by a single walker in a random direction, e.g., with a random angle on a plane in two dimensions. This was later generalized to random flights in three dimensions [3] and RWs on a lattice in d dimensions [4]. A few decades later even more generalized models appeared, e.g., introducing correlation [5–7] or interaction with its past trajectory [8–10], its environment [11–15], or other walkers [16,17]. Despite the plethora of models developed for different applications, still simple isotropic RWs are used as an easy model for Brownian motion and diffusion processes [11,15,18], motion of bacteria [19,20], financial economics [21], detecting community structures in (social) networks [22,23], epidemics [24], polymers in solution [25–27], and home ranges of animals [28,29].

The most important quantity that characterizes RWs is the end-to-end distance and how it scales with the number of steps, giving rise to an exponent ν, i.e., the inverse fractal dimension. To describe the nature of different RW models more thoroughly, other quantities can be used. Here, we are interested in analyzing the “volume” and the “surface” of the RW, which can be conveniently defined by the corresponding quantities of the convex hulls of each given RW. These quantities are used, usually in two dimensions, to describe home ranges of animals [30,31]. But also, very recently, to detect different phases in intermittent stochastic trajectories, like the run and tumble phases in the movement of bacteria [32]. The convex hull of a RW is the smallest convex polytope containing the whole trace of the RW, i.e., it is a nonlocal characteristic that depends on the full history of the walker, namely all visited points.

The most natural statistical observables associated to the convex hull of a random trajectory are its (hyper-) volume and its (hyper-) surface. The full statistics of these two random variables are nontrivial to compute even for a single Brownian motion in two or higher dimensions. Even less is known on the statistics of these two random variables for a discrete-time random walk with a symmetric and continuous jump distributions. In fact, most publications concentrate on the area and perimeter of convex hulls for two-dimensional RWs. The mean perimeter and the mean area of a single random walk in a plane, as a function of the number of steps (in the limit of large number of steps with finite variance of step lengths where it converges to a Brownian motion), are known exactly since more than 20 years [33,34]. These results for the convex hull of a single Brownian motion in a plane have recently been generalized in several directions in a number of studies. These include the exact results for the mean perimeter and mean area of the convex hull for multiple independent Brownian motions and Brownian bridges in a plane [35,36], for the mean perimeter of the convex hull of a single Brownian motion confined to a half plane [37], and for the mean volume and surface of the convex polytopes in arbitrary dimensions d for a single Brownian motion and Brownian bridge [38–40]. Much less is known for discrete-time random walks with arbitrary jump length distributions. Very recently the mean perimeter of the convex hull for planar walks for finite (but large) walk lengths and arbitrary jump distributions were computed explicitly [41]. For the special case of Gaussian jump lengths, an exact combinatorial formula for the mean volume of the convex hull in d-dimensions was recently derived [39]. In $d = 2$, the asymptotic (for large number of steps) behavior of the mean area for Gaussian jump lengths was derived independently in Ref. [41]. Also the convex hulls of other stochastic processes like Lévy flights [42,43], random acceleration processes [44], or branching Brownian motion with absorption [24] were under scrutiny recently.

Analytical calculations of the variance or higher moments turned out to be much more difficult [45,46]. In absence of any analytical result for the full distribution of the volume and surface of the convex hull of a random walk, a sophisticated large-deviation algorithm was recently used to compute numerically the full distribution of the perimeter and the area of the convex hull of a single [47] and multiple [48]

---

[*] hendrik.schawe@uni-oldenburg.de
[†] a.hartmann@uni-oldenburg.de
[‡] satya.majumdar@u-psud.fr

©2017 American Physical Society

random walks in two dimensions. Amazingly, this numerical technique was able to resolve the probability distribution down to probabilities as small as, e.g., 10^{-300} [47,48]. In this work, we will use simulations to obtain the distribution of the volume V and surface ∂V of the convex hull of a single random walk with Gaussian jump length distribution in dimensions $d \in \{3,4\}$ over a large range of its support. In particular, this range is large enough to include large deviations, here down to probability densities far smaller than $P(V) = 10^{-1000}$. While previous work [47,48] suggested that the area and perimeter distribution obeys the large deviation principle in $d = 2$, which was later proven for the perimeter [49], our results suggest that the same holds for higher dimensions. Regarding the scaling behavior of the mean and of the variance, we also study higher dimensions up to $d = 6$. Also we generalize scaling arguments to higher dimensions which were previously used to estimate the properties of these distributions for $d = 2$ [47].

The remainder of the paper is organized as follows. We will first introduce the RW model, give an overview for the calculation of convex hulls in higher dimensions, and describe the sampling technique used to reach the regions of sufficiently small probabilities in Sec. II. The presentation of our results is split into two parts. Section III A compares our numerically obtained means with the analytically derived values from Refs. [38,39] to check that our results are consistent with the literature. Also values for the variances for single and multiple RWs are presented. The behavior of the distributions, especially in their tails, is presented in Sec. III B. Section IV concludes and gives a small outlook to still open questions.

II. MODELS AND METHODS

A. Random walks

A *random walk* [2,4] in d dimensions consists of T step vectors $\boldsymbol{\delta}_i$ such that its position at time τ is given as

$$\boldsymbol{x}(\tau) = \boldsymbol{x}_0 + \sum_{i=1}^{\tau} \boldsymbol{\delta}_i,$$

where $\boldsymbol{x}_0$ is the starting position and chosen in the following always as the origin of the coordinate system. Thus, a realization of a walk can be characterized as a tuple of the displacements $(\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_T)$. We will denote the set of visited points as $\mathcal{P} = \{\boldsymbol{x}(0), \ldots, \boldsymbol{x}(T)\}$. We draw the steps $\boldsymbol{\delta}_i$ from an uncorrelated multivariate Gaussian distribution with zero mean and unit width $G(0,1)$, i.e., d independent random numbers per step. Two examples for dimensions $d = 2$ and $d = 3$ are visualized in Fig. 1. While walks on a lattice show finite-size effects of the lattice structure [47], especially in the region of low probabilities, the Gaussian displacements lead to smooth distributions. Note that in the limit $T \to \infty$ Gaussian and lattice RWs do not behave differently. Both converge to the continuous-time Brownian motion [36].

The RW is very well investigated [1], especially it is known that the end-to-end distance r, and in fact every one-dimensional observable, scales as $r \propto T^{\nu}$ with $\nu = 1/2$. This exponent ν is the same in any dimension and characteristic for diffusion processes.

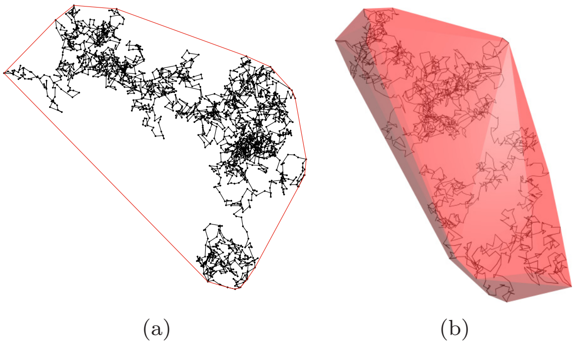

(a) (b)

FIG. 1. Examples for Gaussian random walks in $d = 2$ and $d = 3$. Their convex hull is visualized in red. (a) $d = 2$, $T = 2048$ and (b) $d = 3$, $T = 2048$.

B. Convex hulls

For a given point set $\mathcal{P}$ its *convex hull* $\mathcal{C} = \text{conv}(\mathcal{P})$ is the smallest convex polytope enclosing all points $P_i \in \mathcal{P}$, i.e., all points P_i lie inside the polytope and all straight line segments (P_i, P_j) lie inside the polytope. In Fig. 1 two examples for $d = 2$ and $d = 3$ are shown.

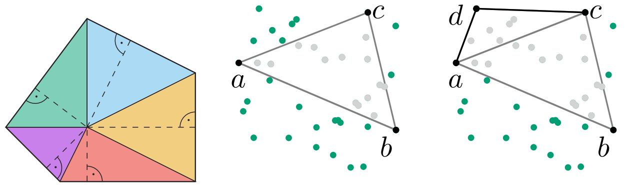
Convex hulls are a well-studied problem with applications from pattern recognition [50] to ecology studies [51]. They are especially important in the context of computational geometry, where next to a wide range of direct applications [52,53] the construction of Voronoi diagrams and Delaunay triangulations [54] stand out, which in turn are useful in a wide range of disciplines [55]. Note that a lower bound for the worst-case time complexity of an exact convex hull algorithm for $T = |\mathcal{P}|$ points is $\Omega(T^{\lfloor d/2 \rfloor})$ [56–58], which is the order of possible facets, i.e., exponential in the dimension. Although, there are approximate algorithms [59,60] that probably would make the examination of higher-dimensional convex hulls feasible, we are only examining the convex hulls up to $d = 6$ using exact algorithms.

We measure the *(hyper-) volume* V, e.g., in $d = 3$ the volume, and the *(hyper-) surface* ∂V, e.g., in $d = 3$ the surface area. Determining surface and volume of a high-dimensional convex polytope is trivial given its facets f_i, which are $(d-1)$-dimensional simplexes. Choosing an arbitrary fixed point p inside the convex polygon, one can create a d-dimensional simplex from each facet f_i, such that their union fills the entire convex hull (cf. Fig. 2(a) for a $d = 2$ example). Therefore, the volume can be obtained by calculating

$$V = \sum_i \text{dist}(f_i, p) a_i / d,$$

where $\text{dist}(f, p)$ is the perpendicular distance from the facet f_i to the point p and a_i is the surface of the facet. The surface of a $(d-1)$-dimensional facet is its $(d-2)$-dimensional volume, which can be calculated with the same method recursively, until the trivial case of one dimensional facets, i.e., lines. Determining the surface uses the same recursion, by calculating $\partial V = \sum_i a_i$.

To foster intuition, this method is pictured for $d = 2$ in Fig. 2(a). Here, the facets are lines and the volume of the simplex is the area of the triangle. The perpendicular distances are visualized as dashed lines.



(a) (b) (c)

FIG. 2. Visualization of (a) the idea to calculate the volume of a convex polygon given its facets and an interior point, perpendicular distances are shown with dashed lines. In (b) and (c) examples of two consecutive recursive steps of the quickhull algorithm are shown. The point d is left of and farthest away from (a,c). Parts of the convex hull are black, discarded points are light gray.

In the scope of this study, we use the *quickhull* algorithm [61–63], and its excellent implementation in the *Qhull* library [64]. Quickhull is a divide-and-conquer algorithm applicable in arbitrary dimensions. For clarity, the algorithm will be explained for $d = 2$, since it makes the central idea clear. The technical details and the generalization to higher dimensions are well explained in Ref. [64].

Start with two points a,b on the convex hull, e.g., the points with minimum and maximum x-coordinate. Determine the point c left (when "looking" $a \to b$) of and farthest away from the edge (a,b) and discard all points inside the polygon (a,b,c). Repeat this step recursively with the edges (a,c) and (c,b) until there are no points on the left side of the current edge. All edges created in this way on the bottom level of the recursion are part of the convex hull. Two steps of this recursion are pictured in Figs. 2(b) and 2(c). The same process is repeated recursively with the point c' left of and farthest away from the inverse edge (b,a).

C. Sampling

We performed Markov chain Monte Carlo simulations to examine the distributions of the volume V and the surface ∂V of the convex hull of RWs in dimensions $d \in \{3,4\}$. To collect *large-deviation* statistics, i.e., obtain not only the peak, but also the tails of the distribution, we use both the classic *Wang-Landau* (WL) sampling [65,66] and a modified Wang-Landau sampling [67–69] with a subsequent entropic sampling [70,71] run. In contrast to similar studies [47,48] no temperature-based sampling scheme was used, since the difficulties to find suitable temperatures and regarding equilibration mentioned in Ref. [47] are even worse in higher dimensions.

Both sampling techniques generate Markov chains of *configurations*, where here configurations are realizations, each given by the tuple of RW displacements $(\boldsymbol{\delta}_1,..,\boldsymbol{\delta}_T)$. One only needs a function yielding an "energy" of a configuration and a way to change a configuration to a similar configuration. As energy we simply use the observable of interest S, i.e., either the volume V or the surface ∂V. To change a configuration, we replace a randomly chosen step $\boldsymbol{\delta}_i$ of the RW with a new randomly drawn step. Because all points $\boldsymbol{x}(\tau)$ for $\tau \geqslant i$ change, this is a global change of the walk. Though, this does not lead to a severe computational overhead, because after the update the convex hull has to be calculated again from scratch in any case.

For both WL versions at first a lower and upper bound of the observable S needs to be defined and the range in between is subdivided in overlapping *windows*, depending on system size T. While the windows can introduce errors in the results, which can lead to neighboring windows that overlap does not match, this phenomenon was not observed in this study. Also, small systems in low dimensions were sampled using the temperature-based method from Ref. [47], which does not use windows, and showed no noteworthy deviations from either of the WL variations. Also, for the present work it was sufficient to sample each window independently in parallel. Therefore, it was not necessary to apply a replica-exchange enhancement [72].

In the beginning, we start with an arbitrary configuration c_i of the walk. Afterwards we repeatedly propose random changes each leading to a new configuration c_{i+1} and accept each with the Metropolis acceptance probability,

$$p_{\mathrm{acc}}[S(c_i) \to S(c_{i+1})] = \min\left(\frac{g[S(c_i)]}{g[S(c_{i+1})]},1\right), \qquad (1)$$

where g is an estimate for the density of states—basically the wanted distribution. If g equals the true density of states this will result in every S being visited with the same probability, i.e., a flat histogram of S. Since we do not know the true density of states in advance, WL iteratively improves the estimate g. Therefore, every time a value of S is visited, $g(S)$ is increased. The original article suggests to multiply $g(S)$ with a fixed factor f to perform the increase, i.e., $g(S) \mapsto g(S)f$, and after an auxiliary histogram fulfills some flatness criterion reduce this factor $f \mapsto \sqrt{f}$. This is repeated until f falls below some beforehand defined threshold f_{final} (here, $f_{\mathrm{final}} = 10^{-8}$). Since the acceptance ratio changes during the simulation, detailed balance does not hold, such that systematic errors are introduced. To mitigate this, a better schedule to modify g is introduced in [68], which reduces the systematic errors. Basically, the flatness criterion is removed and the factor by which to increase $g(S)$ when visiting S is a function of the *Monte Carlo time* t of the simulation, i.e., $\ln[g(S)] \mapsto \ln[g(S)] + t^{-1}$. The sampling terminates as soon as as $t^{-1} \leqslant f_{\mathrm{final}}$ (here, $f_{\mathrm{final}} = 10^{-5}$). This has the added benefit that the simulation time does not depend on some flatness criterion, which is hard to predict, but is at most $1/f_{\mathrm{final}}$ Monte Carlo sweeps.

To remove the systematic error completely, one can use entropic sampling [70,71], i.e., fix the so far obtained estimate g and sample the system using the same acceptance as before from Eq. (1). The entropic sampling pass, which due to the fixed g obeys detailed balance and is thus not subject to the systematic error of the WL sampling, will calculate corrections for the initial estimate g. If the estimate g is close enough to the density of states, it should be able to mitigate the (small) systematic error. Finally, one creates a histogram H of the visited S to arrive at a corrected $\tilde{g}(S) = g(S)H(S)/\langle H\rangle$ [71], where $\langle H\rangle$ is the average number of counts of the histogram.

During this simulation, the value S of the configuration may not leave its window, thus changes to configurations outside of the window are rejected. This also means that the first configuration must be within the window and is therefore obtained via a greedy heuristic. The final distribution is obtained as follows: For mutually overlapping windows, the corresponding densities are multiplied by factors such that in the overlapping regions the densities agree as much as possible. Finally, the density obtained in this way is normalized yielding the whole distribution. To estimate the errors of the distribution, this simulation is done a couple of times and the standard error of the single bins is used as an error estimate.

For the results, which we will present in the following section, we used data from both sampling techniques and in some cases merged them. Comparisons of both techniques showed that the errors introduced by WL have no considerable influence on our results (not shown).

For the determination of mean and variances of convex hull volume and surface the contribution of the tails are negligible, thus we used simple sampling, which enables the simulation of longer walks, i.e., larger values of T, in a larger range of dimensions $d = 2, \dots, 6$,

III. RESULTS

A. Mean and variance

At first, we will verify our simulations by comparing with some analytically known results [33,38] for the mean volume V and surface ∂V scaled appropriately as $\mu_V = \langle V \rangle / T^{d\nu}$ and $\mu_{\partial V} = \langle \partial V \rangle / T^{(d-1)\nu}$. The scaling comes from the $r \propto T^\nu$ scaling of the RW end-to-end, distance, with $\nu = \frac{1}{2}$, in combination with the typical scaling $V \propto r^d$ and $\partial V \propto r^{d-1}$. For large T it is known [38] that

$$\mu_V^\infty = \left(\frac{\pi}{2}\right)^{d/2} \Gamma\left(\frac{d}{2}+1\right)^{-2}, \tag{2}$$

$$\mu_{\partial V}^\infty = \frac{2(2\pi)^{(d-1)/2}}{\Gamma(d)}. \tag{3}$$

This simulation uses simple sampling to sample 10^6 (fewer for $d = 6$ resulting in larger uncertainties) sufficiently long walks from $T_{\min} = 128$ up to $T_{\max} = 262\,144$.

There is an exact result for the mean Volume of the convex hull for finite T [39]:

$$\langle V \rangle = \frac{2^{-d/2}}{\Gamma(d/2+1)} \sum_{n_1,\dots,n_d} \frac{1}{\sqrt{n_1 \dots n_d}} I(n_1,\dots,n_d), \tag{4}$$

where $1 \leqslant n_i \leqslant T$ are integers and

$$I(n_1,\dots,n_d) = \begin{cases} 1 & \text{if } n_1 + \dots + n_d \leqslant T \\ 0 & \text{else .} \end{cases}$$

For example, for $d = 2$ and $d = 3$ this results in

$$\langle V_2 \rangle = \frac{1}{2} \sum_{i=1}^{T} \sum_{j=1}^{T-i} \frac{1}{\sqrt{ij}}, \tag{5}$$

$$\langle V_3 \rangle = \frac{2^{3/2} \times 4}{3\sqrt{\pi}} \sum_{i=1}^{T} \sum_{j=1}^{T-i} \sum_{k=1}^{T-i-j} \frac{1}{\sqrt{ijk}}, \tag{6}$$

respectively. The number of elements in the sums grows with $O(T^d)$ in the number of steps T and the dimension d, such that a numerical evaluation is only feasible for rather small T and d. We calculated some exact values to ensure the quality of our simulations and the extrapolation. These are marked with crosses in Fig. 3.

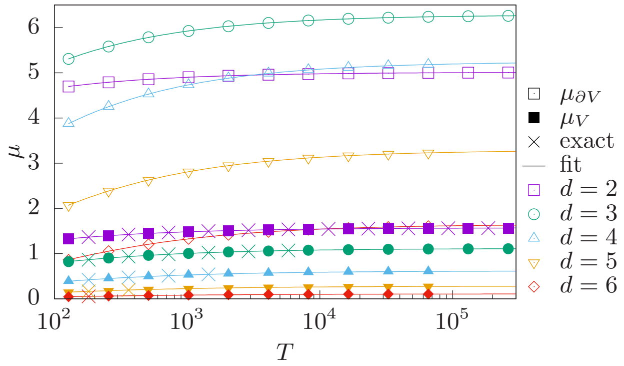


FIG. 3. Scaled mean of the surface $\mu_{\partial V} = \langle \partial V \rangle / T^{(d-1)\nu}$ (open symbols) and volume $\mu_V = \langle V \rangle / T^{d\nu}$ (solid symbols) for different dimensions (different shapes) and walk lengths T obtained by 10^6 samples each. Lines are fits [cf. Eq. (8)] to extrapolate for $T \to \infty$. Crosses are exact values [cf. Eq. (4)] and show very good agreement with the extrapolation. The asymptotic values are shown in Table I. Fits disregard small walk lengths for higher dimensions, since the expansion is valid for large T. To be precise, the fit ranges are $d \geqslant 5$: $T \geqslant 256$ for the surface and $d \geqslant 5$: $T \geqslant 512$ for the volume. They are chosen such that χ^2_{red} reaches a plateau, i.e., does not change significantly if even larger T are ignored (same ranges for the variances). The goodness of fit χ^2_{red} is between 0.3 and 1.2 for all fits. Error bars are smaller than the line of the fit.

To estimate the $T \to \infty$ limit asymptotic value μ_V^∞, it is necessary to extrapolate measurements for different lengths T of the walk. Recently in Ref. [41], the asymptotic large T expansion of the mean area of the convex hull of a 2D Gaussian random walk was worked out explicitly. For Gaussian jump distribution with zero mean and unit variance, it was found that the mean area $\langle A \rangle$ of the convex hull of a walk of T steps has the asymptotic expansion for large T,

$$\frac{\langle A \rangle}{T} = \frac{\pi}{2} + \gamma\sqrt{8\pi}\, T^{-1/2} + \pi(1/4 + \gamma^2)\, T^{-1} + o(T^{-1}), \tag{7}$$

where the constant $\gamma = \zeta(1/2)/\sqrt{2\pi} = -0.58259\dots$. This exact result in 2D leads to a natural guess in higher dimensions for the asymptotic large T expansion of the mean volume of the convex hull, namely,

$$\frac{\langle V \rangle}{T^{d\nu}} = \mu_V + C_1\, T^{-1/2} + C_2\, T^{-1} + o(T^{-1})\,. \tag{8}$$

This guess produces very good fits, shown in Fig. 3, with values in very good agreement with the expectations. We use the same function for the surface and the variances. Though small values of T need to be excluded from the fits, especially for high dimensions. The precise fit ranges are listed in the caption of Fig. 3.

The obtained asymptotic values are listed in Table I. Mind, that the error estimates are purely statistical and do not take into account higher order terms than those present in Eq. (8). To make matters worse, not the same large system sizes could be reached for higher dimensions due to the exponentially increasing time complexity [56].

TABLE I. Analytically expected (top, rounded to four decimal places) and from measurements extrapolated (bottom) asymptotic mean and variance of volume, respectively, surface. Analytical values for the variances are unknown (except for Brownian bridges [46]). Though for the perimeter ($d = 2$) rigorous bounds [73] are known ${\sigma_{\partial V}^{\infty}}^2 \in [2.65 \times 10^{-3}, 9.87]$. Error estimates for the last column are obtained by Gaussian error propagation.

d	μ_V^∞	$\mu_{\partial V}^\infty$	${\sigma_V^\infty}^2$	${\sigma_{\partial V}^\infty}^2$	$\frac{\sigma_V^\infty}{\mu_V^\infty}$
2	1.5708	5.0132			
3	1.1140	6.2832			
4	0.6168	5.2499			
5	0.2800	3.2899			
6	0.1077	1.6493			
2	1.5705(3)	5.0127(5)	0.3078(3)	1.077(1)	0.3532(2)
3	1.1139(2)	6.2832(9)	0.1778(2)	3.093(3)	0.3785(2)
4	0.6164(1)	5.2473(10)	0.05882(7)	2.808(3)	0.3932(2)
5	0.2801(1)	3.2909(9)	0.01274(2)	1.279(2)	0.4032(3)
6	0.1077(1)	1.6492(6)	0.00193(1)	0.351(1)	0.4080(5)

Also, we looked at the average volume $\mu_V = \langle V \rangle / T^{d\nu}$ and surface $\mu_{\partial V} = \langle \partial V \rangle / T^{(d-1)\nu}$ of the convex hulls of multiple RWers with $n \in \{2,3,10,100\}$ independent RWs in $d = 3$ dimensions, which are tabulated in Table II. We determined the listed values in the same way as before with a fit to Eq. (8) (no figure shown) within the same ranges as single walks.

Since the single steps $\boldsymbol{\delta}_i$ are independent, two walkers, i.e., the $n = 2$ case, can be joined at the origin to one walk with twice the number of steps [74], thus $\mu_{V_2}^\infty = 2^{d\nu} \mu_V^\infty$ and $\mu_{\partial V_2}^\infty = 2^{(d-1)\nu} \mu_{\partial V}^\infty$ are the exact mean values for this case. The numerical data is within statistical errors compatible with this expectation. Though, for $n > 2$ this is not as easy anymore. We are not aware of any other published expectations for $d \geqslant 3$.

We have performed the same analysis (no figure shown) for the variances $\sigma_V^2 = \mathrm{Var}(V)/T^{2d\nu}$ and $\sigma_{\partial V}^2 = \mathrm{Var}(\partial V)/T^{2(d-1)\nu}$ and the same remarks apply.

For the ratio between standard deviation and mean,

$$\lim_{d \to \infty} \frac{\sigma_V^\infty}{\mu_V^\infty} = 0$$

is conjectured [38]. Our data shows no downward trend for this ratio as shown in the last column of Table I.

The argument of Ref. [38] is that the expectation of the second moment factorizes to the square of the first moment, if all facets are orthogonal to each other. Since in high dimensions random vectors are with very high probability almost orthogonal, this suggests that the difference of these quantities, which is the variance, should be far smaller than each of them, i.e., the squared mean. Though, in the dimensions under scrutiny, i.e., $d \leqslant 6$ this effect seems not to dominate, because the facets have nonnegligible parallel components.

TABLE II. Analytically expected (top) and from measurements extrapolated (bottom) mean and variance of the volume, respectively, surface of the convex hull of n independent RWs in $d = 3$ dimensions. Analytical values for the variances are unknown. The quality of fit χ^2_{red} for all fits is between 0.4 and 1.7.

n	μ_V^∞	$\mu_{\partial V}^\infty$	${\sigma_V^\infty}^2$	${\sigma_{\partial V}^\infty}^2$
2	3.151	12.566		
2	3.153(1)	12.572(2)	1.427(1)	12.40(1)
3	5.332(1)	17.644(2)	3.796(4)	21.66(2)
10	17.695(2)	37.528(3)	22.54(3)	48.65(4)
100	66.233(7)	85.563(5)	68.65(10)	56.44(7)

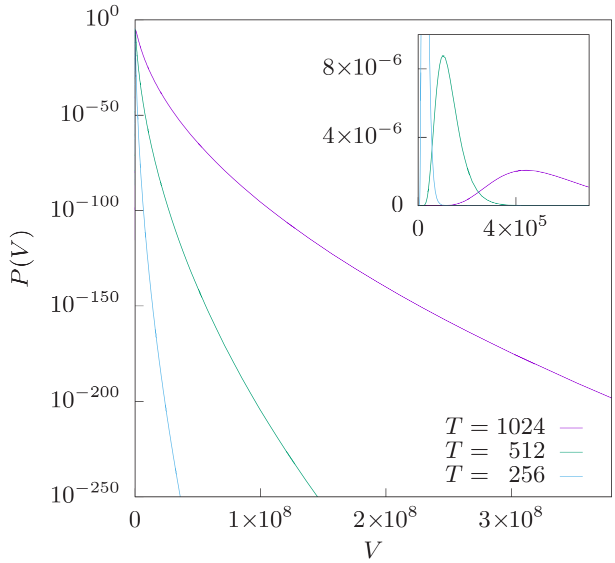


FIG. 4. Distribution of the volume of a $d = 4$ RW for different system sizes T. The inset shows the peak region in linear scale.

It is, of course, hard to estimate for which dimension the orthogonality starts to dominate. As a crude non-rigorous argument we take the scalar product of two random normalized vectors, which approximate the normal vectors of the facets. While its mean value is zero, its variance is $v = 1/d$. This variance is a measure for how parallel the two vectors are. Intuitively, it is clear that in $d = 2$ ($v = 1/2$) and $d = 3$ ($v = 1/3$) most facets have quite large and certainly not negligible parallel components. Then, for $d = 6$ the variance of $v = 1/6$ is not significantly smaller. We would assume that the factorization could dominate the other effects if the parallel component is far smaller—say, 1/20 or 1/100.

Therefore, to draw any conclusions, one should gather results for $d \gg 6$, which may be possible using some fast approximation scheme for convex hulls in high dimensions, though this is beyond the scope of this study.

B. Distributions

In addition to the first moments shown in the previous section, here we look at the actual distribution over a large part of the support. Since the Gaussian distribution, from which the steps are drawn, is not bounded, V and ∂V of a walk consisting of such steps are not bounded, either. Therefore, not the whole support, but a reasonably large part is sampled. Especially, it is large enough to investigate the large-deviation properties of the distribution. As an example, a part of the distribution for the volume of a convex hull of RWs in $d = 4$ dimensions is shown in Fig. 4.

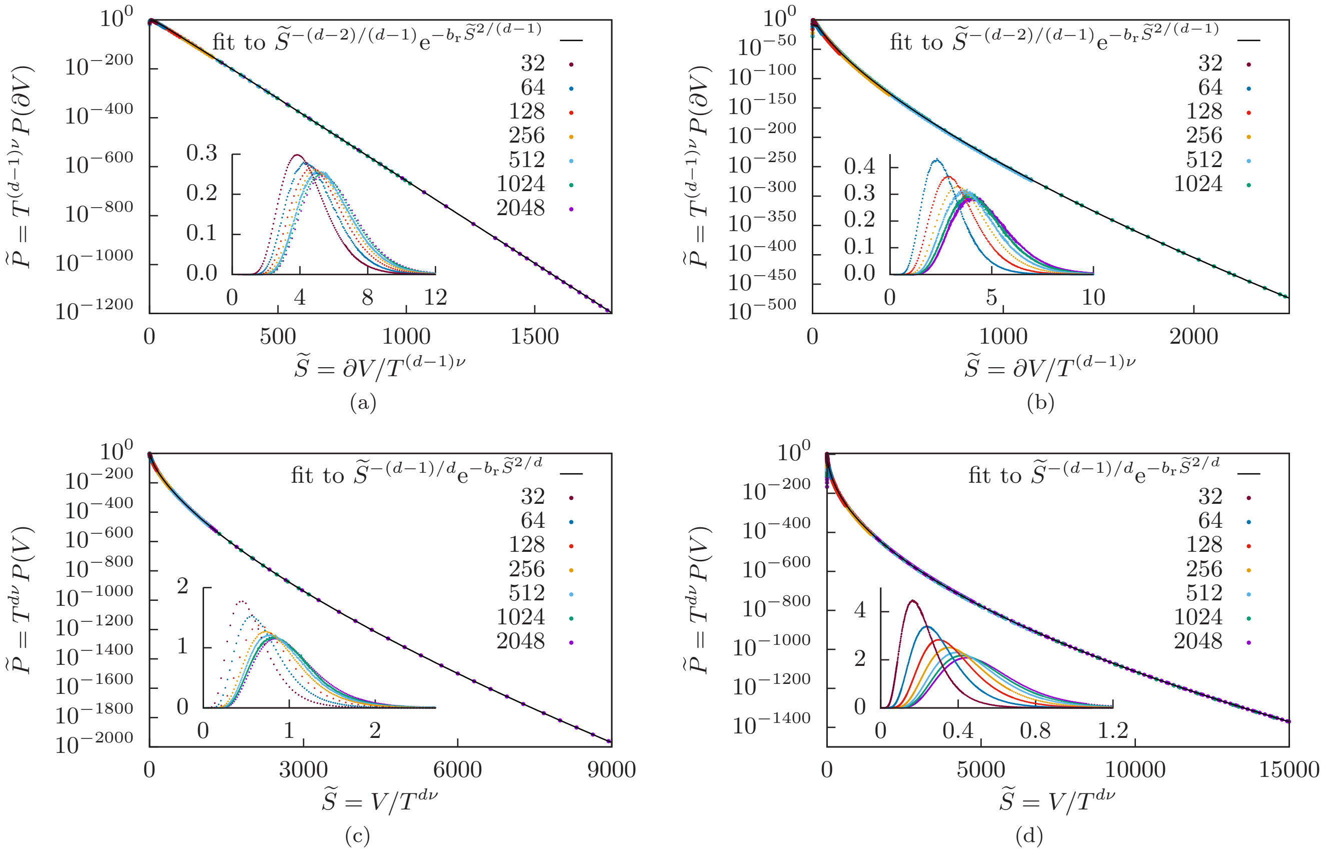


FIG. 5. Distributions of the surface (top) and volume (bottom) for $d \in \{3,4\}$ scaled according to Eq. (9). Statistical errors are smaller than the symbols. The scaling indeed collapses the distributions on one scaling function $\widetilde{P}$. Fits are shown for the largest system size. The inset shows the peak region in linear scale. For larger values of T the collapse works better. (Only a small fraction of all data points are visualized.) (a) $d = 3$, $\widetilde{S} > 500$, $b_r = 1.56$, $\chi^2_{\text{red}} = 2.5$, (b) $d = 4$, $\widetilde{S} > 200$, $b_r = 6.33$, $\chi^2_{\text{red}} = 1.2$, (c) $d = 3$, $\widetilde{S} > 500$, $b_r = 10.61$, $\chi^2_{\text{red}} = 0.8$, and (d) $d = 4$, $\widetilde{S} > 2500$, $b_r = 26.60$, $\chi^2_{\text{red}} = 1.1$.

As we mentioned in the previous section, $\langle V \rangle$ and $\langle \partial V \rangle$ scale for large values of T as $T^{d_e \nu}$ where d_e is the effective dimension of the observable, i.e., $d_e = d$ for the volume and $d_e = d - 1$ for the surface. A natural question is, if the whole distribution does scale according to $T^{d_e \nu}$. Reference [47] already shows that this is true for $d = 2$. For higher dimension we arrive analogously at the scaling assumption for the distribution of the observable S,

$$P(S) = T^{-d_e \nu}\widetilde{P}(ST^{-d_e \nu}). \tag{9}$$

Figure 5 shows the distributions of the volume and surface of the convex hulls of RWs in $d \in \{3,4\}$ dimensions scaled according to Eq. (9). Apparently the scaling works very well in the right tail of larger than typical V. The inset shows that in the peak region there are major corrections to the assumed scaling for small values of T, but it also shows that those corrections rapidly get smaller for larger values of T. A power-law fit with offset to the position of the maxima of the distributions (no figure) with increasing walk length T, confirms convergence for large values of T, i.e., the peaks do collapse on one universal curve for $T \to \infty$.

In fact, the scaling for the distribution of the *span* s, which is the distance between the leftmost and rightmost point, of a one dimensional Brownian motion is known [1,75] to be

$$P(s,T) = (4DT)^{-\nu} f\left(\frac{s}{(4DT)^{\nu}}\right),$$

with some diffusion constant D and

$$f(x) = \frac{8}{\sqrt{\pi}} \sum_{m=1}^{\infty} (-1)^{m+1} m^2\, e^{-m^2 x^2},$$

which has the following asymptotic behavior [47]:

$$f(x) = 2\pi^2 x^{-5}\, e^{-\pi/4x^2}, \qquad \text{for } x \to 0,$$

$$f(x) = \frac{8}{\sqrt{\pi}}\, e^{-x^2}, \qquad \text{for } x \to \infty.$$

With this known $d = 1$ result for the span s, we can construct a guess for the higher dimensional observables, like the volume. Since the one-dimensional projection of a high dimensional RW has the same properties as a one dimensional RW (for this Gaussian model), we use the naive approach $S \propto s^{d_e}$, e.g., $V \approx s^d$. Substituting this into the known result leads to a guess for the expected behavior of the tails with

$$\widetilde{P}(\widetilde{S}) \propto \widetilde{S}^{-(d_e+4)/d_e}\, e^{-b_l \widetilde{S}^{-2/d_e}}, \qquad \text{for } \widetilde{S} \to 0, \tag{10}$$

$$\widetilde{P}(\widetilde{S}) \propto \widetilde{S}^{-(d_e-1)/d_e}\, e^{-b_r \widetilde{S}^{2/d_e}}, \qquad \text{for } \widetilde{S} \to \infty, \tag{11}$$

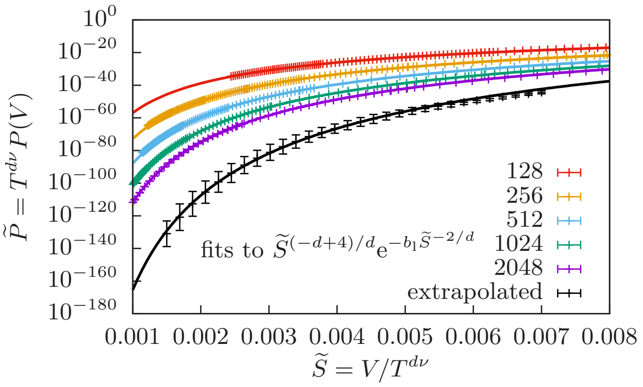


FIG. 6. Fit of the exponential Eq. (10) to the left tail. Note, that the prefactor is constant in this case (for clarity, only every tenths data point is plotted).

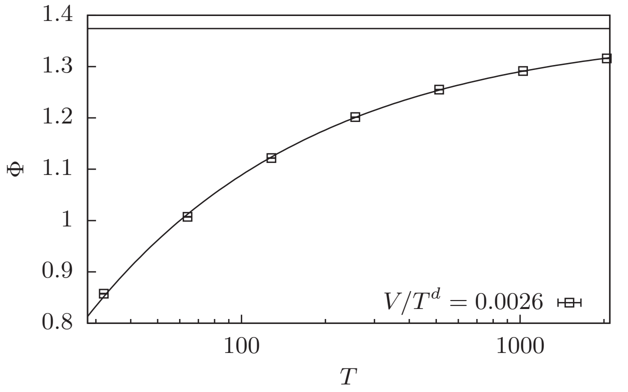


FIG. 7. Point-wise extrapolation of the value of the rate function at a fixed value V/T^d to $T \to \infty$ with a power law, here for a $d =$ 4-dimensional volume. The power-law fit seems to be a reasonable approximation.

where a rescaled $\widetilde{S} = ST^{-\nu d_e}$ is introduced for clarity and with free parameters b_l and b_r. The $\frac{ds}{dS} \propto S^{-(d_e-1)/d_e}$ factors are introduced by the substitution.

For all values of T, the expected distribution for the left tail Eq. (10) fits well to the sampled data, shown for the example of the volume in $d = 4$ in Fig. 6. We extrapolated the curve point-wise to $T \to \infty$ assuming a power-law scaling, resulting in the limit curve in Fig. 6. Similar to the main region of the distribution (shown in Fig. 5), smaller values of T show larger deviations from the limit curve. Note that also the limiting curve fits Eq. (10) (with a suitable values for b_l and the prefactor).

The same analysis for the right tails is shown in Fig. 5, where Eq. (11) is fitted to the right tail of the distributions of the volume and surface in $d \in \{3,4\}$. The good χ^2_{red} values suggest that this is a good estimate of the asymptotic behavior indeed.

To determine whether a distribution P satisfies the large deviation principle [76], i.e., whether it scales as

$$P_T \approx e^{-T\Phi} \tag{12}$$

for some large parameter T, we look if the *rate function* Φ does exist in the $T \to \infty$ limit [76]. Comparing Eq. (12) to the behavior of the right tail [cf. Fig. 5 and Eq. (11)], the rate function seems to be a power law with an exponent $\kappa = 2/d_e$, i.e.,

$$\Phi(S) \propto S^\kappa = S^{2/d_e}. \tag{13}$$

Since we have numerical results for the distribution P, we can determine an empirical rate function Φ of the volume/surface S by extrapolation, (cf. Fig. 7) of

$$\Phi(S/S_{\max}) = -\frac{1}{T}\ln P(S/S_{\max}) \tag{14}$$

to the large T limit. While Φ is usually normalized to $\Phi \in [0,1]$, here S and thus Φ is not bounded. To get a rate function Φ comparable to other publications, we assume $S_{\max} = T^{d_e}$ like Ref. [47]. For the extrapolation we take values of different walk lengths T at multiple values of $S/S_{\max}$, e.g., V/T^d in Fig. 7. These can be thought of as vertical slices through the plot shown in Fig. 8. We use the measured values of Φ to extrapolate it point-wise to $T \to \infty$ using a power law with offset as shown in Fig. 7. Note that since for different walk lengths T we used different histogram bins, we obtain the intermediate values between the discrete bins by cubic spline interpolation. The extrapolation leads to an asymptotic rate function estimate. This shows that the rate function exists and this distributions satisfies the large-deviation principle. This holds for $d = 3$ and $d = 4$, for both volume and surface.

Fitting the power law Eq. (13) through the extrapolated points, as shown in Fig. 8 for the distribution of the volume in $d = 4$, confirms the expectation of $\kappa = 2/d_e$. This holds also for the other cases we considered (not shown as a figure). All measured values of κ are tabulated in Table III and are in reasonable agreement with the expectations. Further, the good quality of the fit χ^2_{red} and the good agreement of the exponents with the expectations, suggests that the power law is a reasonable ansatz and systematic errors due to deviations from this power law or finite-size effects are minor. Hence the given statistical errors should be reasonable. Since the same arguments are applicable for multiple walks, this procedure is

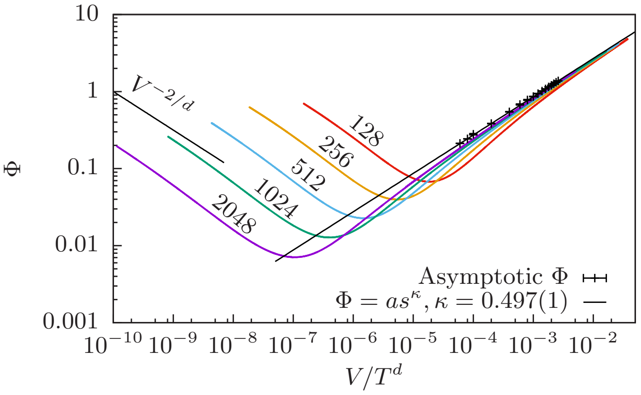


FIG. 8. Rate function of the distribution of the $d =$ 4-dimensional volume of the convex hull of RWs for different walk lengths T. Crosses mark the $T \to \infty$ extrapolated values of the asymptotic rate function as shown in Fig. 7. To those a power law is fitted yielding an estimate for the rate function consistent with the guess in Eq. (13). Further, the expected power law behavior of the left tail is approached.

TABLE III. Comparison of expected and measured rate function exponent κ.

d	Volume V		Surface ∂V	
	Expected κ	Measured κ	Expected κ	Measured κ
2	1	0.994(4)	2	1.996(2)
3	2/3	0.665(1)	1	0.994(2)
4	1/2	0.497(1)	2/3	0.647(5)

tested for the distributions of $m = 3$ multiple walkers in $d = 3$ dimensions, which does also yield within errorbars the same exponent $\kappa = 0.642(17)$ as for the single walker (no figure).

Also note that the power-law relation for the left tail becomes visible, in the far left tail. The expected slope of the left tail $\Phi \propto s^{-2/d}$ [cf. Eq. (10)] is visualized in the far left tail in Fig. 8 and seems to be a reasonable approximation.

IV. CONCLUSIONS

We studied the volume and surface of convex hulls of RWs in up to $d = 6$ dimensions for which we confirmed the analytically known asymptotic means and we estimated the asymptotic variances.

Further, using sophisticated large-deviation sampling techniques we obtained large parts of the distributions P in up to $d = 4$ dimensions down to probability densities far smaller than $P = 10^{-1000}$. The distributions collapse over large ranges of the support (right tail) onto a single curve when being rescaled with the asymptotic behavior of the means. For the left tail, we observe a convergence to a limiting function. Even more, we used our results to confirm the expected functional shapes of the distributions in the left and the right tails, for finite and extrapolated values of T, respectively.

We used simple arguments and numerical simulations to determine the scaling behavior, as well as the asymptotic behavior also for both tails of the rate function $\Phi_r(S) \propto S^{2/d_e}$ for $d \in \{3,4\}$ and are confident that it is valid in arbitrary dimensions.

For future studies, it would be interesting to investigate the properties of the convex hulls of other types of random walks, exhibiting non-trivial values of ν, like self-avoiding walks or loop-erased RWs.

ACKNOWLEDGMENTS

We thank T. Dewenter and S. Zambare for interesting discussions and performing some preliminary test simulations. This work was supported by the German Science Foundation (DFG) through Grant No. HA 3169/8-1. H.S. and A.K.H. thank the LPTMS for hospitality and financial support during one- and two-month visits, respectively, where a considerable part of the projects were performed. The simulations were performed at the HPC clusters HERO and CARL, both located at the University of Oldenburg (Germany) and funded by the DFG through its Major Research Instrumentation Programme (INST Grant No. 184/108-1 FUGG and INST Grant No. 184/157-1 FUGG) and the Ministry of Science and Culture (MWK) of the Lower Saxony State.

---

[1] B. D. Hughes, *Random Walks and Random Environments* (Clarendon Press, Oxford, 1996).

[2] K. Pearson, Nature **72**, 294 (1905).

[3] J. W. Strutt, London Edinburgh Dublin Philos. Mag. J. Sci. **37**, 321 (1919).

[4] G. Pólya, Math. Ann. **84**, 149 (1921).

[5] C. S. Patlak, Bull. Math. Biophys. **15**, 311 (1953).

[6] P. M. Kareiva and N. Shigesada, Oecologia **56**, 234 (1983).

[7] P. Bovet and S. Benhamou, J. Theor. Biol. **131**, 419 (1988).

[8] N. Madras and G. Slade, *The Self-avoiding Walk* (Springer, New York, NY, 2013), Chap. Analysis of Monte Carloimethods, pp. 281–364.

[9] G. F. Lawler, Duke Math. J. **47**, 655 (1980).

[10] A. Weinrib and S. A. Trugman, Phys. Rev. B **31**, 2993 (1985).

[11] M. V. Smoluchowski, Ann. Phys. **353**, 1103 (1916).

[12] W. Alt, J. Math. Biol. **9**, 147 (1980).

[13] P. J. van Haastert and M. Postma, Biophys. J. **93**, 1787 (2007).

[14] B. Weesakul, Ann. Math. Stat. **32**, 765 (1961).

[15] M. Kac, Am. Math. Mon. **54**, 369 (1947).

[16] M. E. Fisher, J. Stat. Phys. **34**, 667 (1984).

[17] G. Schehr, S. N. Majumdar, A. Comtet, and J. Randon-Furling, Phys. Rev. Lett. **101**, 150601 (2008).

[18] T. A. Witten and L. M. Sander, Phys. Rev. B **27**, 5686 (1983).

[19] D. W. Schaefer, Science **180**, 1293 (1973).

[20] E. A. Codling, M. J. Plank, and S. Benhamou, J. R. Soc., Interface **5**, 813 (2008).

[21] E. F. Fama, Financ. Anal. J. **21**, 55 (1965).

[22] M. Rosvall and C. T. Bergstrom, Proc. Natl. Acad. Sci. USA **105**, 1118 (2008).

[23] P. Gupta, A. Goel, J. Lin, A. Sharma, D. Wang, and R. Zadeh, in *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13 (ACM, New York, NY, 2013), pp. 505–514.

[24] E. Dumonteil, S. N. Majumdar, A. Rosso, and A. Zoia, Proc. Natl. Acad. Sci. USA **110**, 4239 (2013).

[25] W. Kuhn, Kolloid-Zeitschrift **68**, 2 (1934).

[26] E. Helfand, J. Chem. Phys. **62**, 999 (1975).

[27] C. Haber, S. A. Ruiz, and D. Wirtz, Proc. Natl. Acad. Sci. USA **97**, 10792 (2000).

[28] F. Bartumeus, M. G. E. da Luz, G. M. Viswanathan, and J. Catalan, Ecology **86**, 3078 (2005).

[29] L. Börger, B. D. Dalziel, and J. M. Fryxell, Ecol. Lett. **11**, 637 (2008).

[30] B. J. Worton, Biometrics **51**, 1206 (1995).

[31] L. Giuggioli, J. R. Potts, and S. Harris, PLoS Comput. Biol. **7**, e1002008 (2011).

[32] Y. Lanoiselée and D. S. Grebenkov, Phys. Rev. E **96**, 022144 (2017).

[33] L. T. Gérard Letac, Am. Math. Mon. **87**, 142 (1980).

[34] G. Letac, J. Theoret. Probabil. **6**, 385 (1993).

[35] J. Randon-Furling, S. N. Majumdar, and A. Comtet, Phys. Rev. Lett. **103**, 140602 (2009).

[36] S. N. Majumdar, A. Comtet, and J. Randon-Furling, J. Stat. Phys. **138**, 955 (2010).

[37] M. Chupeau, O. Bénichou, and S. N. Majumdar, Phys. Rev. E **91**, 050104 (2015).

[38] R. Eldan, Electron. J. Probab. **19**, 1 (2014).

[39] Z. Kabluchko and D. Zaporozhets, Trans. Amer. Math. Soc. **368**, 8873 (2016).

[40] V. Vysotsky and D. Zaporozhets, arXiv:1506.07827 (2015).

[41] D. S. Grebenkov, Y. Lanoiselée, and S. N. Majumdar, J. Stat. Mech. (2017) 103203.

[42] J. Kampf, G. Last, and I. Molchanov, Proc. Am. Math. Soc. **140**, 2527 (2012).

[43] M. Luković, T. Geisel, and S. Eule, New J. Phys. **15**, 063034 (2013).

[44] A. Reymbaut, S. N. Majumdar, and A. Rosso, J. Phys. A: Math. Theor. **44**, 415001 (2011).

[45] T. L. Snyder and J. M. Steele, Proc. Am. Math. Soc. **117**, 1165 (1993).

[46] A. Goldman, Probab. Theory Relat. Fields **105**, 57 (1996).

[47] G. Claussen, A. K. Hartmann, and S. N. Majumdar, Phys. Rev. E **91**, 052104 (2015).

[48] T. Dewenter, G. Claussen, A. K. Hartmann, and S. N. Majumdar, Phys. Rev. E **94**, 052120 (2016).

[49] A. Akopyan and V. Vysotsky, arXiv:1606.07141 (2016).

[50] R. Duda, P. Hart, and D. Stork, *Pattern Classification* (Wiley, New York, 2012).

[51] W. K. Cornwell, D. W. Schwilk, and D. D. Ackerly, Ecology **87**, 1465 (2006).

[52] F. P. Preparata and M. I. Shamos, Convex hulls: Basic algorithms, in *Computational Geometry: An Introduction* (Springer, New York, NY, 1985), pp. 95–149.

[53] M. Jayaram and H. Fleyeh, Amer. J. Intell. Syst. **6**, 48 (2016).

[54] K. Q. Brown, Info. Process. Lett. **9**, 223 (1979).

[55] F. Aurenhammer, ACM Comput. Surv. **23**, 345 (1991).

[56] V. Klee, Archiv der Mathematik **34**, 75 (1980).

[57] R. Seidel, A convex hull algorithm optimal for point sets in even dimensions, Ph.D. thesis, University of British Columbia, 1981.

[58] K. L. Clarkson and P. W. Shor, Discrete Comput. Geom. **4**, 387 (1989).

[59] Z.-B. Xu, J.-S. Zhang, and Y.-W. Leung, Appl. Math. Comput. **94**, 193 (1998).

[60] T. L. V. Hossein Sartipizadeh, Computing the approximate convex hull in high dimensions, (2016), arXiv:1603.04422.

[61] W. F. Eddy, ACM Trans. Math. Softw. **3**, 398 (1977).

[62] A. Bykat, Info. Process. Lett. **7**, 296 (1978).

[63] E. Mücke, Comput. Sci. Eng. **11**, 54 (2009).

[64] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, ACM Trans. Math. Softw. **22**, 469 (1996).

[65] F. Wang and D. P. Landau, Phys. Rev. Lett. **86**, 2050 (2001).

[66] F. Wang and D. P. Landau, Phys. Rev. E **64**, 056101 (2001).

[67] B. J. Schulz, K. Binder, M. Müller, and D. P. Landau, Phys. Rev. E **67**, 067102 (2003).

[68] R. E. Belardinelli and V. D. Pereyra, Phys. Rev. E **75**, 046701 (2007).

[69] R. E. Belardinelli and V. D. Pereyra, J. Chem. Phys. **127**, 184105 (2007).

[70] J. Lee, Phys. Rev. Lett. **71**, 211 (1993).

[71] R. Dickman and A. G. Cunha-Netto, Phys. Rev. E **84**, 026701 (2011).

[72] T. Vogel, Y. W. Li, T. Wüst, and D. P. Landau, Phys. Rev. Lett. **110**, 210603 (2013).

[73] A. R. Wade and C. Xu, Stoch. Processes Appl. **125**, 4300 (2015).

[74] A. Engel, personal communications (2016).

[75] A. Kundu, S. N. Majumdar, and G. Schehr, Phys. Rev. Lett. **110**, 220602 (2013).

[76] H. Touchette, Phys. Rep. **478**, 1 (2009).

A.2. Large deviations of convex hulls of self-avoiding random walks

The first author, Hendrik Schawe, is the author of the thesis at hand. Alexander K. Hartmann is the supervising professor of H. Schawe. Satya N. Majumdar is directeur de recherche at the Laboratoire de Physique Théorique et Modèles Statistiques (LPTMS) at the Université Paris–Sud in Orsay, France.

This publication is a follow-up project to References [39, 40] and Article A.1 which belong to the same DFG grant HA 3169/8-1. All three were limited to normal random walks without memory. The publication at hand treats self-avoiding walks, which necessitates more sophisticated methods to generate samples and uses a different large deviation sampling scheme. This means that the rest of the code for the simulations (mostly the geometric part to calculate the convex hulls) was reused from the previous project. The remaining parts of the simulation and evaluation programs were written by H. Schawe. Similar to the previous project, during frequent meetings of H. Schawe with A. K. Hartmann and some meetings with S. N. Majumdar during a one month stay at the LPTMS, the state and target of the project and possible interesting quantities were discussed. The first draft was prepared by H. Schawe with direct feedback from A. K. Hartmann. In this stage S. N. Majumdar gave some ideas for further improvements, which were incorporated.

PHYSICAL REVIEW E **97**, 062159 (2018)

Large deviations of convex hulls of self-avoiding random walks

Hendrik Schawe[*] and Alexander K. Hartmann[†]
Institut für Physik, Universität Oldenburg, 26111 Oldenburg, Germany
and LPTMS, CNRS, Université Paris-Sud, Université Paris-Saclay, 91405 Orsay, France

Satya N. Majumdar[‡]
LPTMS, CNRS, Université Paris-Sud, Université Paris-Saclay, 91405 Orsay, France

(Received 9 April 2018; revised manuscript received 18 June 2018; published 29 June 2018)

A global picture of a random particle movement is given by the convex hull of the visited points. We obtained numerically the probability distributions of the volume and surface of the convex hulls of a selection of three types of self-avoiding random walks, namely, the classical self-avoiding walk, the smart-kinetic self-avoiding walk, and the loop-erased random walk. To obtain a comprehensive description of the measured random quantities, we applied sophisticated large-deviation techniques, which allowed us to obtain the distributions over a large range of support down to probabilities far smaller than $P = 10^{-100}$. We give an approximate closed form of the so-called large-deviation rate function Φ which generalizes above the upper critical dimension to the previously studied case of the standard random walk. Further, we show correlations between the two observables also in the limits of atypical large or small values.

DOI: 10.1103/PhysRevE.97.062159

I. INTRODUCTION

The standard random walk is a simple Markovian process which has a history as a model for diffusion. There are many exact results known [1]. If memory is added to the model, e.g., to interact with the past trajectory of the walk, analytic treatment becomes much harder. A class of self-interacting random walks that we will focus on in this study are *self-avoiding* random walks, which live on a lattice and do not visit any site twice. This can be used to model systems with excluded volume, e.g., polymers whose single monomers cannot occupy the same site at once [2]. There are more applications which are not as obvious, e.g., a slight modification of the *smart-kinetic self-avoiding walk* traces the perimeter of critical percolation clusters [3], while the *loop-erased random walk* can be used to study spanning trees [4] (and vice versa [5]).

One of the central properties of random walk models is the exponent ν, which characterizes the growth of the end-to-end distance r with the number of steps T, i.e., $r \propto T^{\nu}$. While this has the value $\nu = 1/2$ for the standard random walk, its value is larger for the self-avoiding variations, which are effectively pushed away from their past trajectory. In two dimensions, this value (and other properties) can often be obtained by the correspondence to Schramm-Loewner evolution [6–9]. But between two dimensions and the upper critical dimension, above which the behavior is the same as the standard random walk, Monte Carlo simulations are used to obtain estimates for the exponent ν.

Here we want to study the convex hulls of a selection of self-avoiding walk models featuring larger values of ν. The convex hull allows one to obtain a global picture of the space occupied by a walk without exposing all details of the walk. As an example, convex hulls are used to describe the home ranges of animals [10–12] or the spatial extent of animal epidemics [13]. In physics, they have been proposed to be applied for the analysis of surface diffusion or the detection of binding of molecules [14]. Here, more fundamentally, we will look at the *smart-kinetic self-avoiding walk* (SKSAW), the classical *self-avoiding walk* (SAW), and the *loop-erased random walk* (LERW), since they span a large range of ν values and are well established in the literature. About the convex hulls of standard random walks, we already know plenty of properties. The mean perimeter and area have been known exactly for over 20 years [15,16] for large walk lengths T, i.e., the Brownian motion limit. Since then simpler and more general methods were devised based on Cauchy's formula which relates the support function of a curve to the perimeter and the area enclosed by the curve [17,18]. More recently, also the mean hypervolume and surface for arbitrary dimensions was calculated [19]. For discrete-time random walks with jumps from an arbitrary distribution, the perimeters of the convex hull for finite (but large) walk lengths T were computed explicitly [20]. For the case of Gaussian jump lengths, even an exact combinatorial formula for the volume in arbitrary dimensions is known [21]. For the variance there is an exact result for Brownian bridges [22]. Concerning the full distributions, no exact analytical results are available. Here sophisticated large-deviation simulations were used to numerically explore a large part of the full distribution, i.e., down to probabilities far smaller than 10^{-100} [23–25]. Numerical studies of this kind, which are able to obtain the distribution over a wide range including the extreme tails, are useful to check predictions about, e.g.,

---

[*] hendrik.schawe@uni-oldenburg.de
[†] a.hartmann@uni-oldenburg.de
[‡] satya.majumdar@u-psud.fr

©2018 American Physical Society

large-deviation rate functions such as discussed in Ref. [26]. Furthermore, they can explore new territory and stimulate other studies of large-deviation properties. For example Ref. [23] shows numerical results that the distribution of area and perimeter of the convex hulls of planar standard random walks obey the large-deviation principle, which was later proven by Ref. [27].

Despite this increasing interest in the convex hulls of standard random walks, there seem to be no studies treating the convex hulls of self-avoiding walks. To fill this void, we use Markov chain Monte Carlo sampling to obtain the distributions of some quantities of interest over their whole support. To connect to previous studies [23–25] we also compare the aforementioned variants to the standard random walk on a square lattice (LRW). We are mainly interested in the full distribution of the area A and the perimeter L of $d = 2$ dimensional hulls for walks in the plane, since the effects of the self-interactions are stronger in lower dimensions, but we will also look into the volume V in the $d = 3$ dimensional case. In the past study on standard random walks [25] we found that the full distribution can be scaled to a universal distribution using only the exponent ν and the dimension for large walk lengths T. For the present case, where a walk might depend on its full history, one could expect a more complex behavior. Nevertheless, our results presented below show convincingly that also for self-interacting walks the distributions are universal and governed mainly by the exponent ν, except for some finite-size effects, which are probably caused by the lattice structure. Furthermore, we use the distributions to obtain empirical large-deviation rate functions [28], which suggests that a limiting rate function is mathematically well defined. We also give an estimate for the rate function, which is compatible with the known case of standard random walks and with all cases under scrutiny in this study.

II. MODELS AND METHODS

This section gives a short overview over the models and methods used, with references to literature more specialized on the corresponding subject. Where we deem adequate, also technical details applicable for this study are mentioned.

A. Sampling scheme

To generate the whole distribution of the area or perimeter of the convex hull of a random walk over its full support, a sophisticated Markov chain Monte Carlo (MCMC) sampling scheme is applied [29,30]. The Markov chain is here a sequence of different walk configurations. The fundamental idea is to treat the observable S, i.e., the perimeter, area, or volume, as the energy of a physical system which is coupled to a heat bath with adjustable "temperature" Θ and to sample its equilibrium distribution using the Markov chain. This can be easily done using the classical Metropolis algorithm [31]. Therefore the current walk configuration is changed a bit. (The precise type of change is dependent on the type of walk we are looking at and is explained in the following sections.) The changes must be designed in a way that any configuration can be reached from any other configuration in finite time, i.e., ergodicity must be given. The change is accepted with the acceptance probability

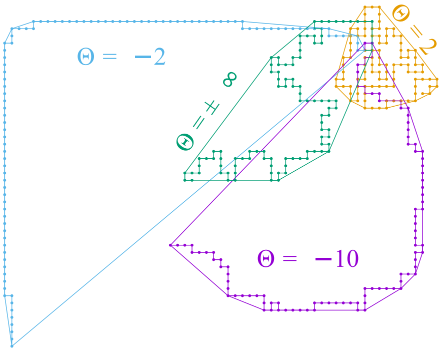


FIG. 1. Typical SAW configurations with $T = 200$ steps and their convex hulls at different temperatures Θ. $\Theta = \pm\infty$ corresponds to a typical configuration without bias.

$$p_{\mathrm{acc}} = \min\{1, \mathrm{e}^{-\Delta S/\Theta}\} \tag{1}$$

and rejected otherwise, which fulfills detailed balance. This means, at long times the Markov process yields configurations $\mathcal{C}$ from its equilibrium distribution $Q_\Theta(\mathcal{C}) = \frac{1}{Z(\Theta)} Q(\mathcal{C})\, \mathrm{e}^{-S(\mathcal{C})/\Theta}$, with the partition function $Z(\Theta)$. For the distribution of the observable $P(S)$ this means

$$P_\Theta(S) = \sum_{\{\mathcal{C}|S(\mathcal{C})=S\}} Q_\Theta(\mathcal{C}) \tag{2}$$

$$= \sum_{\{\mathcal{C}|S(\mathcal{C})=S\}} \frac{\exp(-S/\Theta)}{Z(\Theta)} Q(\mathcal{C}) \tag{3}$$

$$= \frac{\exp(-S/\Theta)}{Z(\Theta)} P(S). \tag{4}$$

That means the "temperature" Θ will bias the configuration towards specific ranges of the "energy" S. Configurations at small and negative Θ will show larger than typical S; small and positive Θ show smaller than typical S and large values independent of the sign sample configurations from the peak of the distribution. Figure 1 shows typical walk configurations of the self-avoiding walk at different values of Θ.

In a second step, histograms of the equilibrium distribution $P_\Theta(S)$ are corrected for the bias introduced via Θ. Using Eq. (4), we can easily remove this bias and arrive at the unbiased distribution

$$P(S) = \mathrm{e}^{S/\Theta} Z(\Theta) P_\Theta(S). \tag{5}$$

The free parameter $Z(\Theta)$ can be obtained by enforcing continuity and normalization of the distribution. This necessitates that we perform this sampling at multiple Θ such that there are good statistics over the whole range and overlapping histograms from which to choose $Z(\Theta)$, so that the overlapping regions coincide, i.e., the distribution is continuous. This, at the same time, serves as a quality estimate of the Markov process, since the overlaps will only coincide cleanly over their whole range if the samples were taken in equilibrium. So a clean coincidence is a strong hint at a good quality of the data. Further details and examples can be found in several other articles, where it has been applied and explained for specific models [26,29,30,32–35], but also in a very general form [36].

In particular, the algorithm was already used successfully in other studies looking at the large deviation properties of convex hulls of random walks [23,24].

B. Lattice random walk

All of the self-interacting random walks, which are the focus of this study, are typically treated on a lattice. Hence, we will start by introducing the simple, i.e., noninteracting, isotropic random walk on a lattice. For simplicity we will use a square lattice with a lattice constant of 1. A realization consists of T randomly chosen discrete steps $\boldsymbol{\delta}_i$. Here we use steps between adjacent lattice sites, i.e., d-dimensional Cartesian base vectors $\boldsymbol{e}_i$, which are drawn uniformly from $\{\pm\boldsymbol{e}_i\}$. The realization can be defined as the tuple of the steps $(\boldsymbol{\delta}_1,...,\boldsymbol{\delta}_T)$ and the position at time τ as

$$\boldsymbol{x}(\tau) = \boldsymbol{x}_0 + \sum_{i=1}^{\tau} \boldsymbol{\delta}_i. \tag{6}$$

Here we set the start point $\boldsymbol{x}_0$ at the coordinate origin. The set of visited sites is therefore $\mathcal{P} = \{\boldsymbol{x}(0), \ldots, \boldsymbol{x}(T)\}$.

The central quantity of the LRW is the average end-to-end distance

$$r = \sqrt{\langle(\boldsymbol{x}(T) - \boldsymbol{x}_0)^2\rangle}\,, \tag{7}$$

where $\langle\ldots\rangle$ denotes the average over the disorder. It grows polynomially and is characterized by the exponent ν via $r \propto T^{\nu}$. For the LRW it is $\nu = 1/2$, which is typical for all diffusive processes.

As the change move for the Metropolis algorithm, we replace a randomly chosen $\boldsymbol{\delta}_i$ by a new randomly drawn displacement. This way we can clearly reach any possible configurations, i.e., ergodicity holds. Since our quantity of interest is the convex hull, i.e., a global property of the walk, we do not profit much from local moves, e.g., crankshaft moves. Thus we use this simple, global move.

C. Smart-kinetic self-avoiding walk

The smart-kinetic self-avoiding walk [3,37] is probably the most naive approach to a self-avoiding walk. It grows on a lattice and never enters sites it already visited. Since it is possible to get trapped on an island inside already-visited sites, this walk needs to be *smart* enough to never enter such traps.

In $d = 2$ it is possible to avoid traps using just local information in constant time using the *winding angle* method [37]. In conjunction with hash table backed detection of occupied sites, a realization with T steps can be constructed in time $\mathcal{O}(T)$.

This method will typically yield longer stretched walks than the LRW due to the constraint that it needs to be self-avoiding. This can be characterized by the exponent ν, which is larger than $1/2$ in $d = 2$.

The sketch of Fig. 2 shows that this ensemble does not contain every configuration with the same probability but prefers closely winded configurations. This is also visible in Fig. 3(b). This is characterized by the exponent $\nu = 4/7$ [9], which is larger than ν for the LRW but smaller than for the SAW. Also note that it is conjectured that the upper critical dimension is $d = 3$ [37], i.e., $\nu = 1/2$ for all $d \geqslant 3$—possibly with logarithmic corrections in $d = 3$. Therefore only $d = 2$ is simulated in this study.

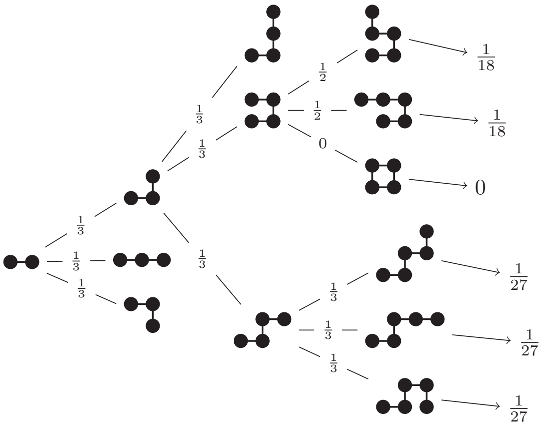
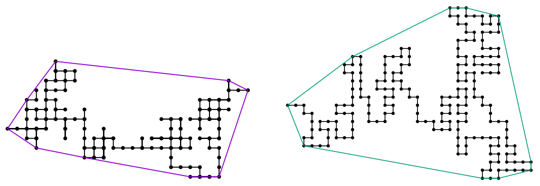
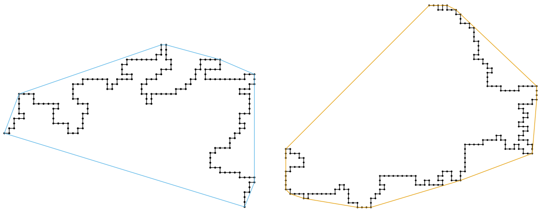


FIG. 2. Decision tree visualizing the probability to arrive at certain configurations following the construction rules of the SKSAW. Not all possible configuration have the same probability, and hence this rule defines a different ensemble than SAW.

While it is easy to draw realizations from this ensemble uniformly, i.e., simple sampling, it is not so straightforward to apply the MCMC changes. If one just changes single steps like for the LRW and accepts if it is self-avoiding or rejects if it is not, one will generate all self-avoiding walk configurations with equal probability. Our approach to generate realizations according to this ensemble handles the construction of the walk as a *black box*. It acts on the random numbers used to generate a realization from scratch. During the MCMC at each iteration one random number is replaced by a new random number and a SKSAW realization is regenerated from scratch using the modified random numbers [36]. Since every configuration of underlying random numbers can occur this way, every possible SKSAW configuration can be constructed, such that this protocol is ergodic. This change is then accepted according to Eq. (1) and undone otherwise.



(a) LRW (b) SKSAW



(c) SAW (d) LERW

FIG. 3. Typical configurations with $T = 200$ steps, drawn uniformly from the corresponding ensembles, of all types of random walks under scrutiny in this study with their convex hulls.

D. Self-avoiding random walk

While the above-mentioned SKSAW does produce self-avoiding walks, SAW denotes another ensemble, the ensemble where realizations are drawn uniformly from the set of all self-avoiding configurations. It is not trivial to sample from this distribution efficiently. The black-box method used for SKSAW is not feasible, since the construction of a SAW takes time exponential in the length with simple methods like dimerization [2,38]. It is possible to perform changes directly on the walk configuration and accept them according to Eq. (1), but their rejection rate is typically quite high and the resulting configurations are very similar [2], which makes this inefficient. The state-of-the-art method to sample SAW is the *pivot algorithm* [2]. It chooses a random point and uses it as the pivot for a random symmetry operation, i.e., rotation or mirroring. If the resulting configuration is not self-avoiding, it is rejected. Otherwise we accept it with the temperature-dependent acceptance probability Eq. (1).

As mentioned previously, the exponent $\nu = 3/4$ [7] is larger than for the SKSAW. Since the upper critical dimension for SAW is $d = 4$, this study will also look at $d = 3$, where an exact value of ν is not known and the best estimate is $\nu = 0.587\,597(7)$ [39], though our focus is on $d = 2$ for this type.

While there are highly efficient implementations of the pivot algorithm [39,40], the time complexity of the problem at hand is dominated by the time needed to construct the convex hull; thus we go with the simple hash-table-based $\mathcal{O}(T)$ approach [2].

E. Loop-erased random walk

The LERW [41] uses a different approach to achieve the self-avoiding property. It is built as a simple LRW, but each time a site is entered for the second time, the loop that is formed, i.e., all steps since the first entering of this site, is erased. While this ensures no crossings in the walk, the resulting ensemble is different from the SAW ensemble and the walks are longer stretched out, as characterized by the larger exponent $\nu = 4/5$ [5,8,42]. Similar to the SAW, the upper critical dimension is $d = 4$ and an estimate for $d = 3$ is $\nu = 0.615\,76(2)$ [43].

For construction—similar to SKSAW—we need to keep all used random numbers and change them in the MCMC algorithm. This leads to a dramatically higher memory consumption than simple sampling, where each loop can be discarded as soon as it is closed.

F. Convex hulls

We will study the *convex hulls* $\mathcal{C}$ of the sites visited by the random walk $\mathcal{P}$. The convex hull of a point set $\mathcal{P}$ is the smallest polytope containing all points $P_i \in \mathcal{P}$ and all line segments (P_i, P_j). Some example hulls are shown in Fig. 3.

Convex hulls are one of the most basic concepts in computational geometry,[1] with noteworthy application in the construction of Voronoi diagrams and Delaunay triangulations [44].

For point sets in the $d = 2$ plane, we use Andrew's *monotone chain* [45] algorithm for its simplicity and *Quickhull* [46] as implemented by QHULL [47] for $d = 3$. Both algorithms have a time complexity of $\mathcal{O}(T \ln T)$. In $d = 2$ Andrew's monotone chain algorithm results in ordered points of the convex hull. Adjacent points (i, j) in this ordering are the line segments of the convex hull. Quickhull results in the simplicial facets of the convex hull.

To obtain the perimeter of a $d = 2$ convex hull, we sum the lengths of its line segments L_{ij}. To calculate the area and the volume, we use the same fundamental idea. In both cases we subdivide the area/volume into simplexes, i.e., triangles for the area and tetrahedra for the volume. Therefore we choose an arbitrary fixed point p_0 inside of the convex hull and construct a simplex for each facet f_m, i.e., for $d = 2$ each line segment of the hull $f_m = (i, j)$ forms a triangle (i, j, p_0), and each triangular face $f_m = (i, j, k)$ of a $d = 3$ dimensional polyhedron forms a tetrahedron with p_0. The volume of a triangle is trivially

$$A_{ijp_0} = \tfrac{1}{2}\,\mathrm{dist}(f_m, p_0) L_{ij},$$

where $\mathrm{dist}(f_m, p_0)$ is the perpendicular distance from a facet f_m to a point p_0. Since the union of all triangles built this way is the whole polygon, the sum of their areas is the area of the polygon. Similarly, the volume of a polyhedron is the sum of the volumes of all tetrahedra constructed from its faces. The volume of the individual tetrahedra is given by

$$V_{ijkp_0} = \tfrac{1}{3}\,\mathrm{dist}(f_m, p_0) A_{ijk}.$$

For random walks on a lattice with T steps of length 1 in d dimensions the maximum volume is

$$S_{\max} = \frac{(T/d_{\mathrm{e}})^{d_{\mathrm{e}}}}{d_{\mathrm{e}}!} \tag{8}$$

for T divisible by the effective dimension d_{e} of the observable, e.g., 2 for the area of a planar hull or 3 for the volume in three dimensions. For example, the configuration of maximum area corresponds to an L shape, i.e., $A_{\max} = \frac{T^2}{8}$. This form can be derived by the general volume of a d-dimensional simplex defined by its $d + 1$ vertices $\boldsymbol{v}_i$ [48]:

$$V = \frac{1}{d!} \det(\boldsymbol{v}_1 - \boldsymbol{v}_0, \ldots, \boldsymbol{v}_d - \boldsymbol{v}_0). \tag{9}$$

Without loss of generality, we set $\boldsymbol{v}_0$ to be the coordinate origin. To achieve maximum volume all $\boldsymbol{v}_i, i > 0$ need to be orthogonal and of equal length. Thus a random walk going T/d steps along some base vector $\boldsymbol{e}_i$ and continuing with T/d steps in direction $\boldsymbol{e}_{i+1}$ has a convex hull defined by the tetrahedron specified by $\boldsymbol{v}_i = \sum_{j=1}^{i} \frac{T}{d} \boldsymbol{e}_j$. The matrix $M = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d)$ is thus triangular and its determinant is the product of its diagonal entries $M_{ii} = \frac{T}{d}$, which leads directly to Eq. (8). An exception occurs in $d = 2$, where the perimeter is $L_{\max} = 2T$.

III. RESULTS

The focus of this work lies on $d = 2$ dimensional SAW and LERW. The results for higher dimensions and for SKSAW are generated with less numerical accuracy. The LRW results also

---

[1] Three of the first four examples for static problems of computational geometry in Wikipedia can utilize convex hulls for their solution (https://en.wikipedia.org/wiki/Computational_geometry, 12.01.2018).

have a lower accuracy, as their purpose is mainly to scrutinize the effect of the lattice structure underlying all considered walk types in comparison to the nonlattice results from [25]. Also, not all combinations are simulated but only those listed with a value in Table II.

The same raw data is evaluated for equidistant bins and logarithmic bins. And the respective variants are shown according to the scaling of the x axis.

A. Correlations

To get an intuition for how the configurations with atypical large areas A or perimeters L look like, we visualize the correlation between these two observables as scatter plots in Fig. 4.

Since the smallest possible SAW is an (almost) fully filled square, there cannot be instances below some threshold, which explains the gaps on the left side of the scatter plots and of the distributions shown in the following section. In the center of the scatter plots, which is already in probability regions far beyond the capabilities of simple sampling methods, the behavior becomes strongly dependent on the bias.

If biasing for large perimeters (top), the area shows a nonmonotonous behavior. First, somehow larger perimeters come along typically with larger areas for entropic reasons, i.e., there are fewer configurations which are long and thin, and more bulky, which have a larger area. Though, for the far-right tail, the only configurations with extreme large perimeters are almost linelike and thus have a very small area. Also note that the excluded volume effect of the SAW leads to overall larger areas at the same perimeters.

On the other hand, when biasing for large areas (bottom) the configurations with largest area, which are L-shaped (cf. Fig. 1), unavoidably have quite large perimeters; hence the scatter plots show an almost linear correlation between area and perimeter. Since the configurations of large areas naturally avoid self-intersections, since steps on already visited points do not enlarge the convex hull, the differences between LRW and SAW diminish in the right tail. Note that with the large-area bias, no walks with the very extreme perimeters exist, for the reason already mentioned.

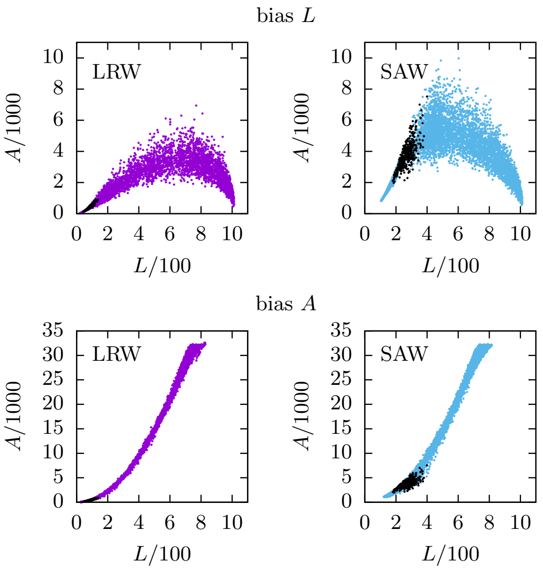


FIG. 4. The top row shows data from simulations biasing towards larger (and smaller) than typical perimeters L. The bottom row biases the area A. The left column shows data from LRW and the right from SAW both with $T = 512$ steps. The results of simple sampling are shown in black. Note that only very narrow parts are covered by simple sampling for the LRW.

Note, however, that these scatter plots are very dependent on which observable we are biasing for. In principle we observe that small perimeters are strongly correlated with small areas, while for large but not too large perimeters, there is a broad range of area sizes possible. For extremely large perimeters, the area must be small. For a comprehensive analysis, one would need a full two-dimensional histogram, which could be obtained using Wang-Landau sampling but which is beyond the scope of this study and would require a much larger numerical effort. Nevertheless, from looking at Fig. 4 one can anticipate that the two-dimensional histogram would exhibit a strong correlation for small values of L and a broad scatter of the accessible values of A for larger but not too large values of L.

B. Moments and distributions

The distributions of the different walk types differ considerably. This can be observed in Fig. 5, where distributions of the area A for all types with $T = 1024$ steps are drawn. The main part of the distribution shifts to larger values for larger values of ν as expected, and the probability of atypically large areas is boosted even more in the tails.

In the right tail, the distributions seem to bend down. Below, where we show results for different walk sizes T, we see that this is a finite-size effect of the lattice structure and the fixed step length. This can be seen also as follows: Since the lattice together with the fixed step length sets an upper bound on the area, the probability plummets near this bound for entropic reasons, i.e., there are for any walk length T only eight configurations with maximum area (due to symmetries) such that all self-avoiding types will meet at this point (not visible because the bins are not fine enough).

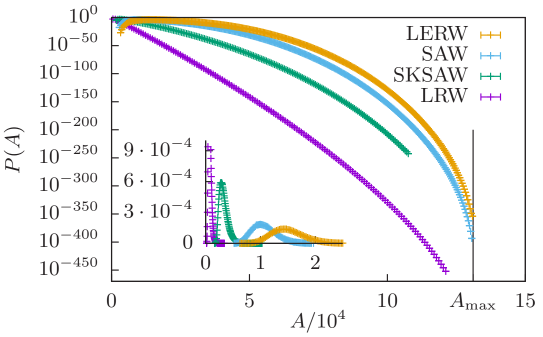


FIG. 5. Distribution of all scrutinized walk types with $T = 1024$ steps. The vertical line at $A_{\max} = 131\,072$ denotes the maximum area [Eq. (8)], i.e., SAW and LERW are sampled across their full support and SKSAW and LRW are not. The inset shows the peak region. The gap on the left is due to excluded volume effects, i.e., there are no configurations with area below some threshold, since this would require self-intersection.

This is supported from Ref. [23], which shows that the distribution $P(A)$ for standard random walks with Gaussian jumps, i.e., without lattice or fixed step length, do not bend down and have an exponential right tail. We conclude that the deviations from this are thus caused by this difference.

First we will look at the rescaled means $\mu_S = \langle S \rangle / T^{d_e \nu}$, where S is an observable and d_e its effective dimension, as introduced above in Eq. (8). The scaling is a combination of the scaling of the end-to-end distance $r \propto T^\nu$ and the typical scaling that a d-dimensional observable scales as r^d with a characteristic length r.

Nevertheless, due to finite-size corrections, the ratios $\mu_S = \langle S \rangle / T^{d_e \nu}$ will still depend on the walk length. Thus, the measured estimates $\mu_S = \mu_S(T)$ at specific walk lengths T need to be extrapolated to get an estimate of the asymptotic value $\mu_S^\infty = \lim_{T \to \infty} \mu_S(T)$. For the extrapolation we use [25]

$$\mu_S(T) = \mu_S^\infty + C_1 T^{-1/2} + C_2 T^{-1} + o(T^{-1}). \tag{10}$$

This choice is motivated by a large-T expansion for the area A ($d_e = 2$) of the convex hulls of standard random walks ($\nu = 1/2$) with Gaussian jumps [20],

$$\frac{\langle A \rangle}{T} = \frac{\pi}{2} + \gamma \sqrt{8\pi}\, T^{-1/2} + \pi(1/4 + \gamma^2)\, T^{-1} + o(T^{-1}), \tag{11}$$

where the constant $\gamma = \zeta(1/2)/\sqrt{2\pi} = -0.582\,59\ldots$. A natural guess for a generalization to observables of a different effective dimension d_e [25] and different walk types would be a similar behavior with different coefficients like Eq. (10).

Indeed, using this form to estimate the asymptotic means μ_S^∞ of the observable S yields good fits, as visible in Fig. 6. In fact, for the fit quality we obtain χ^2_{red} values between 0.4 and 1.7. (The fit ranges for SKSAW begin at $T = 512$, and for LRW, SAW, and LERW at $T = 128$, hinting at more severe corrections to scaling for the former.) We assume that the scaling is thus valid for arbitrary random walk types. The resulting fit parameters are shown in Table I.

For standard random walks with Gaussian jumps the asymptotic means $\mu^\infty_{S,\text{Gaussian}}$ are known [19]. These results can be used to predict the corresponding values for LRW. First consider the following heuristic argument for a $d = 2$ square lattice. On average a random walk takes the same amount of steps in x and y direction such that on average two steps displace the walker by $\sqrt{2}$, i.e., the diagonal of a square. In contrast, a Gaussian walker with variance 1 will be displaced on average by 1 every step. To make both types comparable, we can increase the lattice constant to $\sqrt{2}$, which leads to an average displacement of 1 per step for the LRW. Using the same argumentation for higher dimensions, we can use the trivial scaling with the lattice constant S^{d_e} and the length of the diagonal of a unit hypercube $d^{1/2}$ to derive a general conversion:

$$\mu^\infty_{S,\text{LRW}} = \mu^\infty_{S,\text{Gaussian}} / d^{d_e/2}. \tag{12}$$

These known results are listed next to our measurements in Table I and are within error bars compatible with our measurements.

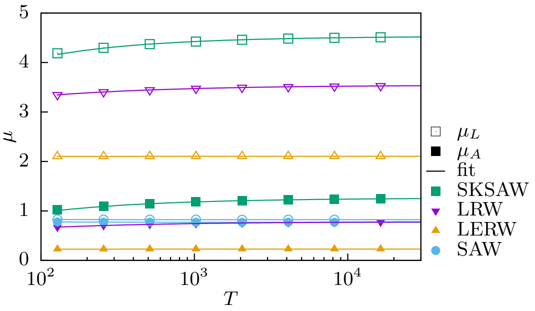


FIG. 6. Scaled means $\mu_A = \langle A \rangle / T^{2\nu}$ and $\mu_L = \langle L \rangle / T^\nu$ for different walk types. The lines are fits to extrapolate the asymptotic values shown in Table I according to Eq. (10). Error bars of the values are smaller than the line of the fit and are not shown for clarity.

TABLE I. Asymptotic mean values extrapolated from simulational data and the exactly known values for the standard random walk (LRW). The columns labeled with μ_L^∞ and μ_A^∞ are for $d = 2$, those labeled with $\mu_{\partial V}^\infty$ and μ_V^∞ are for $d = 3$. For $d = 3$ we did not simulate the SKSAW, see Sec. II C. Also, SAW has lower accuracy because of fewer samples in $d = 3$.

	μ_L^∞	μ_A^∞	$\mu_{\partial V}^\infty$	μ_V^∞
LRW (exact)	3.5449...	0.7854...	2.0944...	0.21440...
LRW	3.5441(7)	0.7852(2)	2.0945(4)	0.21445(4)
SKSAW	4.5355(12)	1.2642(5)		
SAW	0.8233(7)	0.7714(1)	2.070(2)	0.1998(2)
LERW	2.1060(3)	0.2300(1)	1.6436(2)	0.13908(3)

Since we have data for whole distributions, a natural question is whether this scaling does apply over the whole support of the distribution. There is evidence that this is true for the convex hulls of standard random walks [23] in arbitrary dimensions [25]. That means the distributions of an observable S for different walk lengths T should collapse onto one universal function

$$P(S) = T^{-d_e \nu} \widetilde{P}(S T^{-d_e \nu}). \tag{13}$$

Figure 7 shows the distributions of the $d = 2$ area of all considered random walk types scaled according to Eq. (13). The curves collapse well in the peak region and in the intermediate-right tail. In the far-right tail, clear deviations from a universal curve are obvious, which are the mentioned finite-size effects caused by the lattice.

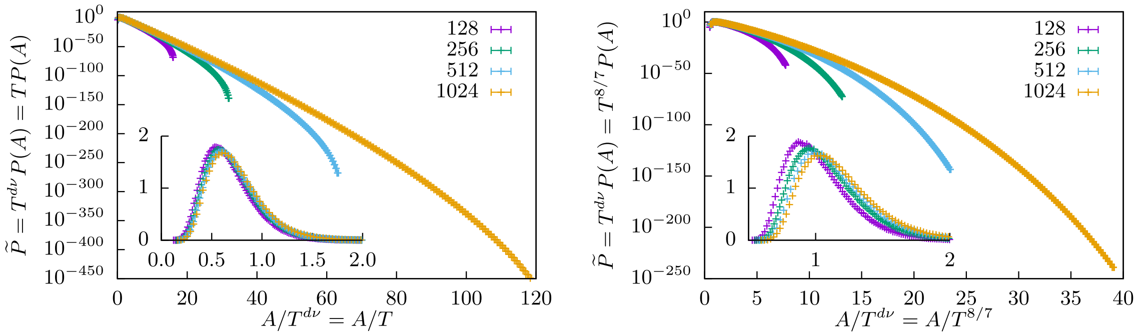
The distributions look qualitatively similar, though with weaker finite-size effects, i.e., a better collapse, for the perimeter L (not shown). In $d = 3$, where we have studied the volume, the results also look similar but exhibit stronger finite-size effects (not shown).

Using the full distributions at different values of the walk length P_T, we can test if it obeys the large-deviation principle, i.e., if Φ exists, such that the distribution scales as

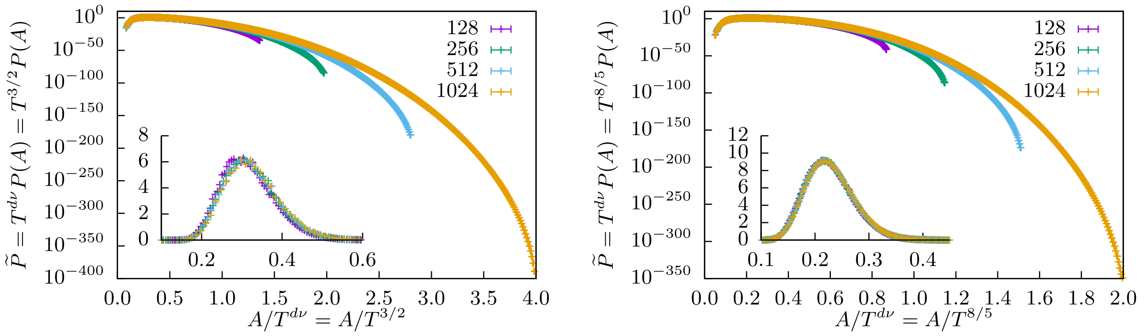
$$P_T \approx \mathrm{e}^{-T\Phi} \tag{14}$$

for large values of T [28]. To simplify comparison, the support of the rate function is usually normalized to [0,1]. Here we achieve this by using the maximum Eq. (8). Solving Eq. (14) for Φ results in

$$\Phi(S/S_{\max}) = -\frac{1}{T} \ln P(S/S_{\max}). \tag{15}$$



(a) LRW (b) SKSAW



(c) SAW (d) LERW

FIG. 7. Distributions of the area of different types of random walks scaled according to Eq. (13) for different walk lengths T.

We plot this for a selection of our results in Fig. 8. From these plots, Φ seems to approximately follow a power law in the intermediate-right tail, while the finite-size effects caused by the lattice play a major role in the far-right tail, which consequently "bends up."

Assuming that the rate function behaves approximately as a power law, which seems consistent with our data shown in Fig. 8, i.e.,

$$\Phi(s) \propto s^{\kappa}, \tag{16}$$

the exponent κ can be estimated by combining the definition of Φ Eq. (14) with the scaling assumption Eq. (13) as follows; note that for clarity we use here $S_{\max} \propto T^{d_e}$:

$$\exp(-T\Phi(S/T^{d_e})) \approx \frac{1}{T^{\nu d_e}}\widetilde{P}(S/T^{\nu d_e}). \tag{17}$$

The $1/T^{\nu d_e}$ term on the right-hand side can be ignored next to the exponential. Since the right-hand side is a function of $S/T^{\nu d_e}$, the left-hand side must also be dependent only on $S/T^{\nu d_e}$. This can be achieved by assuming $-\nu d_e\kappa + d_e\kappa = 1$, as one can easily see by using Eq. (16):

Starting from the left-hand side

$$\begin{aligned}
&\exp(-T^1\Phi(S/T^{d_e}))\\
&\propto \exp(-T^1(S/T^{d_e})^{\kappa})\\
&= \exp(-T^{-\nu d_e\kappa + d_e\kappa}(S/T^{d_e})^{\kappa})\\
&= \exp(-(S/T^{\nu d_e})^{\kappa}).
\end{aligned}$$

From this we can conclude

$$\kappa = \frac{1}{d_e(1-\nu)}, \tag{18}$$

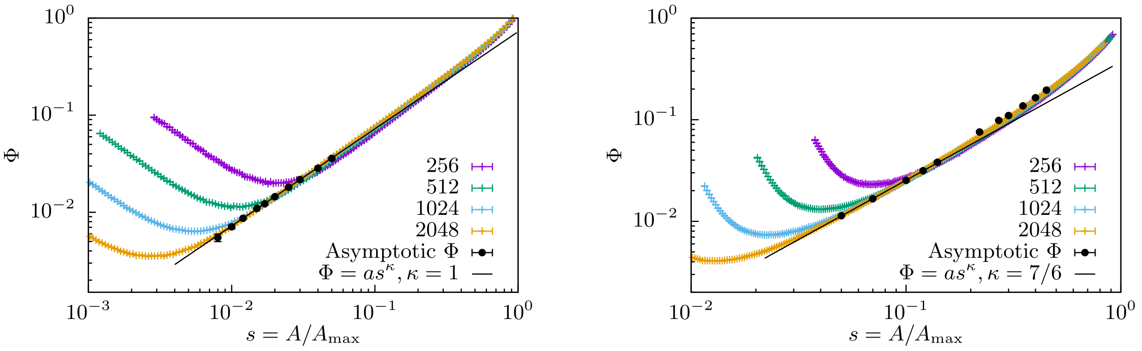
which simplifies to the case of the standard random walk above the critical dimension of the given walk type [25],

$$\kappa_g = \frac{2}{d_e}.$$

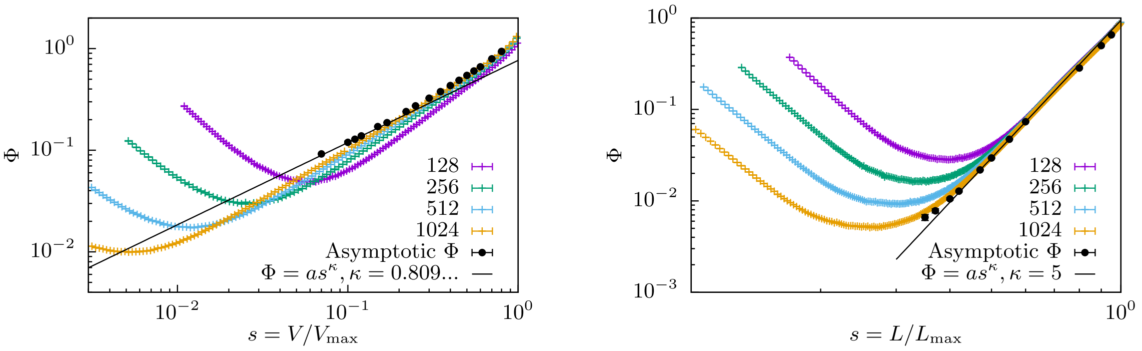
To compare this crude estimate with the results of our simulations, we do a pointwise extrapolation of the empirical rate functions for fixed walk lengths T as done before in Refs. [23–25]. For the pointwise extrapolation, we use measurements Φ_T for multiple values of the walk length T at fixed values of $S/S_{\max}$. Since our data are discrete due to binning, the values of Φ_T are obtained by cubic spline interpolation. With these data points, which can be thought of as vertical slices through the plots of Fig. 8, we extrapolate the $T \to \infty$ case with a fit to a power law with offset

$$\Phi = aT^b + \Phi_{\infty}. \tag{19}$$

The extrapolated values are marked with black dots in Fig. 8. Since finite-size effects have a major impact on the tails due to the lattice structure, we expect that our estimate is only valid for the intermediate right tail of our simulational data. To estimate sensible uncertainties, we fit different ranges of our data and give the center of the range of the obtained κ as our estimate with an error including the extremes of the obtained κ. The black lines in Fig. 8 are our expected values, which are in all examples compatible with some range of our extrapolated data.



(a) LRW (b) SKSAW



(c) SAW (d) LERW

FIG. 8. Selection of asymptotic rate functions extrapolated from our data and our expected exponent κ of the rate function Φ.

All exponents κ we calculated, together with our expectations, are listed in Table II. A more detailed discussion of the examples shown in Fig. 8 follows.

In Fig. 8(a) the LRW is shown, which is equivalent to Brownian motion in the large-T limit for which Refs. [23] and [25] showed the rate function to behave like a power law with exponent $\kappa = 1$ for the area in $d = 2$. Using the above-mentioned procedure we obtain $\kappa = 0.99(2)$, which is in perfect agreement with the expectation $\kappa = 1$.

Figure 8(b) shows the same for the SKSAW. The obtained asymptotic rate function's exponent $\kappa = 1.28(12)$ is compatible with our expectation, though the stronger finite-size effects lead to larger uncertainties of our estimate.

Figure 8(c) shows the same but for the volume of the SAW in $d = 3$ dimensions. The finite-size effects are apparently stronger for the volume in $d = 3$, as the slope of the right-tail rate function gets less steep with increasing system size.

TABLE II. Comparison of expected and measured rate function exponent κ. The value is the center of multiple fit ranges and the error is chosen such that the largest and the smallest result is enclosed.

	V		∂V	
	Eq. (18)	κ	Eq. (18)	κ
LRW	1	0.99(2)	2	
SKSAW	$\frac{7}{6}$	1.28(12)	$\frac{7}{3}$	
SAW	2	2.2(4)	4	4.11(14)
SAW $d = 3$	0.809...	0.92(11)	1.214...	
LERW	$\frac{5}{2}$	2.57(24)	5	4.82(19)
LERW $d = 3$	0.867...	0.89(9)	1.299...	

Figure 8(d) shows the same for the perimeter of a $d = 2$ dimensional LERW. In contrast to the area and volume, the far-right tail of the perimeter seems to bend down instead of up, albeit slightly. Though in the intermediate right tail, the rate function seems to behave as expected.

In general, our data supports the convergence to a limiting rate function, which, mathematically speaking, means that the *large-deviation principle* holds. This means that the distributions are somehow well behaved and might be accessible to analytical calculations, alhough the estimate for what the rate function Φ actually is can possibly be improved. However, since our estimate for κ is always compatible with our measurements it appears plausible that also for interacting walks the distribution of the convex hulls is governed by the scaling behavior of the end-to-end distance, as given by the exponents ν.

IV. CONCLUSIONS

We numerically studied the area and perimeter of the convex hulls of different types of self-avoiding random walks in the plane and to a lesser degree the volume of their convex hulls in $d = 3$ dimensional space. By applying sophisticated large-deviation algorithms, we calculated the full distributions, down to extremely small probabilities like 10^{-400}. We also obtained corresponding rate functions of these observables. Our data support a convergence of the rate functions, which means the large-deviation principle seems to hold. We observed a generalized scaling behavior, which was before established for standard random walks. Thus, although the self-avoiding types of walk exhibit a more complicated behavior as compared to standard random lattice walks, and although the limiting scaled distributions of their convex hull's volume and surface look quite different for the various walk cases, in the end the convex hull behavior seems to be still governed by the single end-to-end distance scaling exponent ν.

We also observed, rather expectedly, that the two observables area and perimeter are highly correlated for small values. For large but not too large values of the perimeter, many different values of the area are possible but are statistically dominated by rather small values of the area. Extremely large values of the perimeter are only feasible with shrinking area.

Finally, we gave estimates for the large-T asymptotic mean values of the mentioned observables. These might be of interest for attempts to calculate these values analytically.

For future studies it could be interesting to look closer into the correlations between different observables that we briefly noted. For a more thorough study, it would be useful to obtain full two-dimensional histograms.

ACKNOWLEDGMENTS

This work was supported by the German Science Foundation (DFG) through Grant No. HA 3169/8-1. H.S. and A.K.H. thank the LPTMS for hospitality and financial support during one- and two-month visits where a considerable parts of the projects were performed. The simulations were performed at the HPC clusters HERO and CARL, both located at the University of Oldenburg (Germany) and funded by the DFG through its Major Research Instrumentation Programme (INST 184/108-1 FUGG and INST 184/157-1 FUGG) and the Ministry of Science and Culture (MWK) of the Lower Saxony State. We also thank the GWDG Göttingen for providing computational resources.

[1] B. D. Hughes, *Random Walks and Random Environments* (Clarendon Press, Oxford, UK, 1996).

[2] N. Madras and G. Slade, Analysis of Monte Carlo methods, in *The Self-Avoiding Walk* (Springer, New York, 2013), pp. 281–364.

[3] A. Weinrib and S. A. Trugman, Phys. Rev. B **31**, 2993 (1985).

[4] S. S. Manna, D. Dhar, and S. N. Majumdar, Phys. Rev. A **46**, R4471 (1992).

[5] S. N. Majumdar, Phys. Rev. Lett. **68**, 2329 (1992).

[6] J. Cardy, Ann. Phys. **318**, 81 (2005).

[7] G. F. Lawler, O. Schramm, and W. Werner, arXiv:math/0204277.

[8] G. F. Lawler, O. Schramm, and W. Werner, Conformal invariance of planar loop-erased random walks and uniform spanning trees, in *Selected Works of Oded Schramm*, edited by I. Benjamini and O. Häggström (Springer, New York, 2011), pp. 931–987.

[9] T. Kennedy, J. Stat. Phys. **160**, 302 (2015).

[10] C. O. Mohr, American Midland Naturalist **37**, 223 (1947).

[11] B. J. Worton, Ecol. Model. **38**, 277 (1987).

[12] S. A. Boyle, W. C. Lourenco, L. R. da Silva, and A. T. Smith, Folia Primatol. **80**, 33 (2009).

[13] E. Dumonteil, S. N. Majumdar, A. Rosso, and A. Zoia, Proc. Natl. Acad. Sci. USA **110**, 4239 (2013).

[14] Y. Lanoiselée and D. S. Grebenkov, Phys. Rev. E **96**, 022144 (2017).

[15] L. T. Gérard Letac, Am. Math. Mon. **87**, 142 (1980).

[16] G. Letac, J. Theoret. Probab. **6**, 385 (1993).

[17] J. Randon-Furling, S. N. Majumdar, and A. Comtet, Phys. Rev. Lett. **103**, 140602 (2009).

[18] S. N. Majumdar, A. Comtet, and J. Randon-Furling, J. Stat. Phys. **138**, 955 (2010).

[19] R. Eldan, Electron. J. Probab. **19**, 1 (2014).

[20] D. S. Grebenkov, Y. Lanoiselée, and S. N. Majumdar, J. Stat. Mech. (2017) 103203.

[21] Z. Kabluchko and D. Zaporozhets, Trans. Am. Math. Soc. **368**, 8873 (2016).

[22] A. Goldman, Probab. Theory Relat. Fields **105**, 57 (1996).

[23] G. Claussen, A. K. Hartmann, and S. N. Majumdar, Phys. Rev. E **91**, 052104 (2015).

[24] T. Dewenter, G. Claussen, A. K. Hartmann, and S. N. Majumdar, Phys. Rev. E **94**, 052120 (2016).

[25] H. Schawe, A. K. Hartmann, and S. N. Majumdar, Phys. Rev. E **96**, 062101 (2017).

[26] A. K. Hartmann, P. L. Doussal, S. N. Majumdar, A. Rosso, and G. Schehr, Europhys. Lett. **121**, 67004 (2018).

[27] A. Akopyan and V. Vysotsky, arXiv:1606.07141.

[28] H. Touchette, Phys. Rep. **478**, 1 (2009).

[29] A. K. Hartmann, Phys. Rev. E **65**, 056102 (2002).

[30] A. K. Hartmann, Eur. Phys. J. B **84**, 627 (2011).

[31] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, J. Chem. Phys. **21**, 1087 (1953).

[32] A. Engel, R. Monasson, and A. K. Hartmann, J. Stat. Phys. **117**, 387 (2004).

[33] A. K. Hartmann, Phys. Rev. Lett. **94**, 050601 (2005).

[34] S. Wolfsheimer, B. Burghardt, and A. K. Hartmann, Algorithms Mol. Biol. **2**, 9 (2007).

[35] P. Fieth and A. K. Hartmann, Phys. Rev. E **94**, 022127 (2016).

[36] A. K. Hartmann, Phys. Rev. E **89**, 052103 (2014).

[37] K. Kremer and J. W. Lyklema, Phys. Rev. Lett. **54**, 267 (1985).

[38] K. Suzuki, Bull. Chem. Soc. Jpn. **41**, 538 (1968).

[39] N. Clisby, Phys. Rev. Lett. **104**, 055702 (2010).

[40] N. Clisby, J. Stat. Phys. **140**, 349 (2010).

[41] G. F. Lawler, Duke Math. J. **47**, 655 (1980).

[42] A. J. Guttmann and R. J. Bursill, J. Stat. Phys. **59**, 1 (1990).

[43] D. B. Wilson, Phys. Rev. E **82**, 062102 (2010).

[44] K. Q. Brown, Inf. Process. Lett. **9**, 223 (1979).

[45] A. Andrew, Inf. Process. Lett. **9**, 216 (1979).

[46] A. Bykat, Inf. Process. Lett. **7**, 296 (1978).

[47] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, ACM Trans. Math. Software **22**, 469 (1996).

[48] P. Stein, Am. Math. Mon. **73**, 299 (1966).

A.3. Large Deviations of Convex Hulls of the “True” Self-Avoiding Random Walk

The first author, Hendrik Schawe, is the author of the thesis at hand. Alexander K. Hartmann is the supervising professor of H. Schawe.

This publication is a follow-up project to References [39, 40] and Articles A.1 and A.2 which belong to the same DFG grant HA 3169/8-1. H. Schawe attended the XXX IUPAP[1] Conference on Computational Physics (CPP 2018) in Davis, CA, USA; this conference offered to publish original work in the accompanying conference proceedings. Therefore, H. Schawe chose to start new simulations in the spirit of Article A.2 to justify a proceedings publication with original material. Fortunately, the “true” self-avoiding random walk model showed some surprising behavior, which makes this publication stand for its own instead of being just a summary of existing works with a slight extension. The first draft was prepared by H. Schawe, which was then refined in multiple iterations by A. K. Hartmann and H. Schawe.

---

[1] International Union of Pure and Applied Physics

arXiv:1808.10698v1 [cond-mat.stat-mech] 31 Aug 2018

Large Deviations of Convex Hulls of the "True" Self-Avoiding Random Walk

Hendrik Schawe, Alexander K Hartmann

Institut für Physik, Universität Oldenburg, 26111 Oldenburg, Germany

E-mail: `hendrik.schawe@uni-oldenburg.de`

Abstract. We study the distribution of the area and perimeter of the convex hull of the "true" self-avoiding random walk in a plane. Using a Markov chain Monte Carlo sampling method, we obtain the distributions also in their far tails, down to probabilities like 10^{-800}. This enables us to test previous conjectures regarding the scaling of the distribution and the large-deviation rate function Φ. In previous studies e.g., for standard random walks, the whole distribution was governed by the Flory exponent ν. We confirm this in the present study by considering expected logarithmic corrections. On the other hand, the behavior of the rate function deviates from the expected form. For this exception we give a qualitative reasoning.

1. Introduction

The random walk is a very simple model for diffusive processes with Brownian motion [1] as the prime example. Though its applications range from financial models [2] over online search engines [3] to the very sampling algorithm used in this study [4]. Its simplest variation lives on a lattice and takes steps on random adjacent sites at each timestep, which is exceptionally well researched [5]. With the further constraint that no site may be visited twice, such that the walk is *self-avoiding*, it becomes a simple model for polymers [6]. Interestingly, depending on the exact protocol how the self-avoidance is achieved, they can also be used to study the perimeter of, e.g., critical percolation clusters [7] or spanning trees [8, 9]

The distance of a random walk from its starting point is the most prominent and simple measurable quantity. Nevertheless, here we go beyond this by considering the convex hull of all T sites visited by the random walk, i.e., the smallest convex polygon containing all these sites. It can be seen as a measure of the general shape of the random walk, without exposing all details of the walk. Thus, the area A or perimeter L of the convex hull can then be used to characterize the random walk in a very simple way. This method is also used, for example, to describe the home ranges of animals [10, 11, 12], spread of animal epidemics [13] or classification of different phases using the trajectory of intermittent stochastic processes [14]. For standard random walks its mean perimeter [15] and mean area [16] in the large T limit are known exactly since a long time. More recently different approaches generalized these results to multiple random walks [17, 18] and arbitrary dimensions [19]. Even more recently the mean perimeter and area for finite (but large) walk lengths T were computed explicitly [20] if the random walk is discrete-time with jumps from an arbitrary distribution. If the distribution of the jump length is Gaussian, even an exact combinatorial formula for the mean volume in arbitrary dimensions is known [21]. For higher moments however, there is only one analytic result for the special case of Brownian bridges [22], i.e., closed walks with Gaussian jumps. When asking for more, i.e., for the full distributions, no exact analytical results are available. This motivated the numerical study of the full distributions—or at least large parts of the support—using large-deviation sampling techniques to sample even far into the tails of standard random walks [23] and multiple random walks [24], also in higher dimensions [25]. These numerical studies are rather expensive, since they usually require Markov chain Monte Carlo simulations, allowing one to measure the distribution in regions where the probabilities are as small as 10^{-100}.

Since self-avoiding walks are considerably more difficult to treat analytically than standard random walks, there are no analytical results about the properties of their convex hulls yet. Therefore, the authors of this contribution very recently published a numerical study of the full distribution of perimeter and area of three different types of self-avoiding random walks [26], notably the classical *self-avoiding walk* (SAW) and the *smart kinetic self-avoiding walk* (SKSAW). While the SAW is combinatorial in nature and describes the set of all self-avoiding configurations with equal probability, the SKSAW is a growth process, which assigns more weight to some configurations. In [26] we also give an estimate for the functional form of the rate function Φ describing the far right tail behavior of the distribution, i.e., $P(S) \approx \mathrm{e}^{-T\Phi(S)}$. It was found to depend only on the dimension d and the scaling exponent ν. For two-dimensional random walks these scaling exponents are often known exactly through Schramm-Loewner evolution [27, 28, 29, 30].

In this study we test this prediction for the *"true" self-avoiding walk* (TSAW), which has a free parameter β governing how strictly self-avoiding the walk is. Introduced in [31] the TSAW was a counter model to the SAW, especially it should demonstrate that the behavior of the combinatorial SAW is very different from more natural growing random walks which avoid themselves. Indeed, in two dimensions, where the end-to-end distance r of a T step SAW scales as $r \propto T^\nu$ with $\nu = 3/4$, the TSAW will scale as

$$r \propto T^\nu (\ln T)^\alpha \tag{1}$$

with $\nu = 1/2$ [32] and a correction α, which is not known rigorously, but estimated as $\alpha = 1/4$ [33]. Here we show, for large-enough values of β, that in contrast to previous work [26] the rate function Φ is not simply determined by the value of ν, since the growth process of the TSAW in the large-area region of the tail is indistinguishable from the SKSAW growth process, although they have different values of the scaling exponent ν determined by the behavior of the high-probability part of their distributions.

2. Models and Methods

This section will introduce the TSAW model and the sampling method in enough detail to reproduce the results of this study. For more fundamental methods, like the construction of the convex hull, we will sketch the main ideas.

2.1. Large Deviation Sampling Scheme

To obtain good statistic in the far tail, it is not sufficient to perform naive simple sampling, since configurations of probability P would need about $1/P$ samples to occur at all. It is therefore not feasible to explore the distributions down to the tails of $P < 10^{-100}$ with simple sampling. Instead we use an importance sampling scheme to generate more samples in the low probability tails. Thus, we generate Markov chains consisting of sequences of TSAWs and use the well known Metropolis algorithm [34] with a Boltzmann sampling weight. For this purpose, we identify the quantity S we are interested in—here the area A but it could be any measurable quantity in principle—with the energy occurring in the Boltzmann factor and introduce an artificial "temperature" Θ. Since the TSAW is a growth process, it is not trivial to come up with a local change move within the Markov chain, i.e., it is difficult to change a configuration by a small amount while preserving the correct statistics. Therefore our Markov chain is not directly a chain of configurations of TSAW but rather a chain of random number vectors $\boldsymbol{\xi}_i$. Each vector $\boldsymbol{\xi}_i$ determines a configuration of a TSAW by performing the growth process and using for each of the T decisions a random number from $\boldsymbol{\xi}_i$. This approach is sketched in Fig. 1 and extremely general since it can be applied to any model [35]. A change move is a simple change of one entry of $\boldsymbol{\xi}_i$.

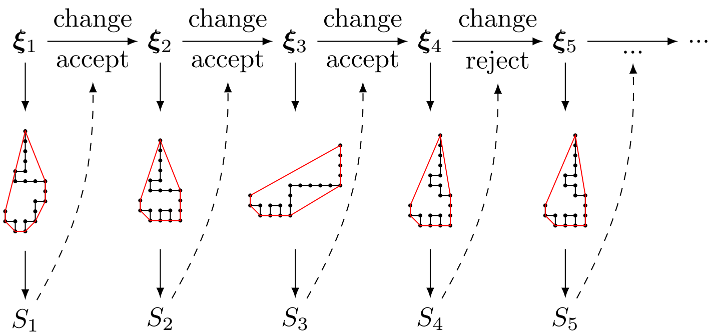


Figure 1. Sketch of the Markov chain of random number vectors $\boldsymbol{\xi}_i$. The change move is performed on the $\boldsymbol{\xi}_i$ and a new TSAW is generated from scratch, its energy difference to the previous configuration is used to accept or reject the change.

Following the Metropolis algorithm, we propose a new $\boldsymbol{\xi}'$ by replacing a random entry with a new random number $\xi \in U[0,1)$, generate a new TSAW configuration from scratch using the random numbers $\boldsymbol{\xi}'$ and calculating its energy S', i.e., its area. The proposed configuration is then accepted, i.e., $\boldsymbol{\xi}_{i+1} = \boldsymbol{\xi}'$, or rejected, i.e., $\boldsymbol{\xi}_{i+1} = \boldsymbol{\xi}_i$, depending on the temperature and energy difference with respect to the previous configuration with probability $p_{\mathrm{acc}} = \mathrm{e}^{-\Delta S/\Theta}$, where $\Delta S = S' - S_i$ is the energy difference caused by the change. Replacing a random entry by a new entry is clearly ergodic, since any possible $\boldsymbol{\xi}_i$ can be generated this way. Since we use the classical Metropolis acceptance probability, detailed balance is also given. This Markov process will therefore yield configurations $\boldsymbol{\xi}$ according to their equilibrium distribution $Q_\Theta(\boldsymbol{\xi}) = \frac{1}{Z(\Theta)} Q(\boldsymbol{\xi})\, \mathrm{e}^{-S(\boldsymbol{\xi})/\Theta}$, where $Q(\boldsymbol{\xi})$ is the natural, unbiased distribution of configurations and $Z(\Theta)$ the corresponding partition function. For small temperatures this will lead to small energies, i.e., smaller than typical perimeters or areas. For large temperatures typical configurations will be generated and for negative temperatures larger than usual energies dominate. Since this Metropolis algorithm will generate instances following a Boltzmann distribution we can easily undo this bias, i.e., we can derive the actual distribution $P(S)$ from the biased, temperature dependent distributions $P_\Theta(S)$ as

$$\begin{aligned} P_\Theta(S) &= \sum_{\{\boldsymbol{\xi}|S(\boldsymbol{\xi})=S\}} Q_\Theta(\boldsymbol{\xi}) && (2)\\ &= \sum_{\{\boldsymbol{\xi}|S(\boldsymbol{\xi})=S\}} \frac{\exp(-S/\Theta)}{Z(\Theta)} Q(\boldsymbol{\xi}) && (3)\\ &= \frac{\exp(-S/\Theta)}{Z(\Theta)} P(S). && (4) \end{aligned}$$

The unknown $Z(\Theta)$ can be numerically determined by enforcing the continuity of the distribution. Therefore we need to simulate the system at many different temperatures Θ, such that all histograms $P_\Theta(S)$ overlap with adjacent temperatures. $Z(\Theta)$ can now be calculated in overlapping regions, which should coincide for continuity, i.e.,

$$\mathrm{e}^{S/\Theta_i}\, Z(\Theta_i) P_{\Theta_i}(S) = \mathrm{e}^{S/\Theta_{i+1}}\, Z(\Theta_{i+1}) P_{\Theta_{i+1}}(S) \tag{5}$$

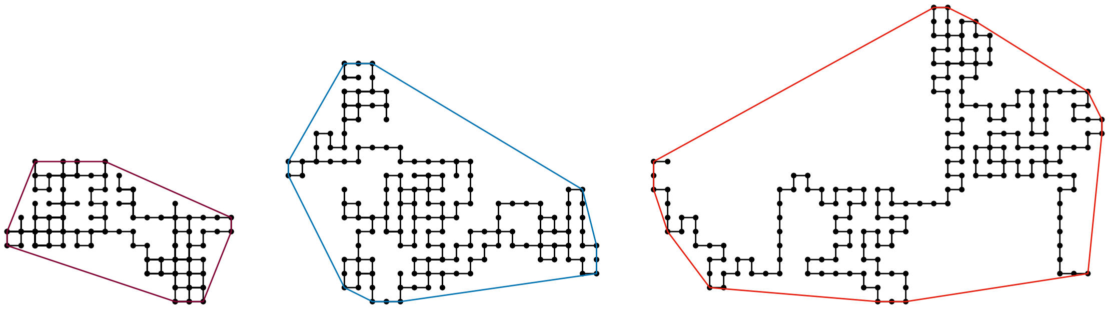
$$\Rightarrow \quad \frac{Z(\Theta_i)}{Z(\Theta_{i+1})} = \exp\left(S/\Theta_{i+1} - S/\Theta_i\right) \frac{P_{\Theta_{i+1}}(S)}{P_{\Theta_i}(S)}. \tag{6}$$

This relation fixes all ratios of consecutive $Z(\Theta)$. The absolute value can be fixed by the normalization of the whole distribution.

This method is applicable to a wide range of models, and already successfully applied to obtain, e.g., the distributions over a large range for the score of sequence alignments [36, 37, 38], work distributions for non-equilibrium systems [35], properties of Erdős Rényi random graphs [39, 40, 41], and in particular to obtain statistics of the convex hulls of a wide range of types of random walks [23, 24, 26].

2.2. "True" Self-Avoiding Walk

The "true" self-avoiding Walk (TSAW) is a random walk model, in which the walker tries to avoid itself, but self-avoidance is not strictly imposed. To construct a TSAW realization one grows a standard random walk on a lattice and records the number of visits n_i to each site i. For each step the probability to step on a neighboring site i is weighted with the number of times that site was already visited

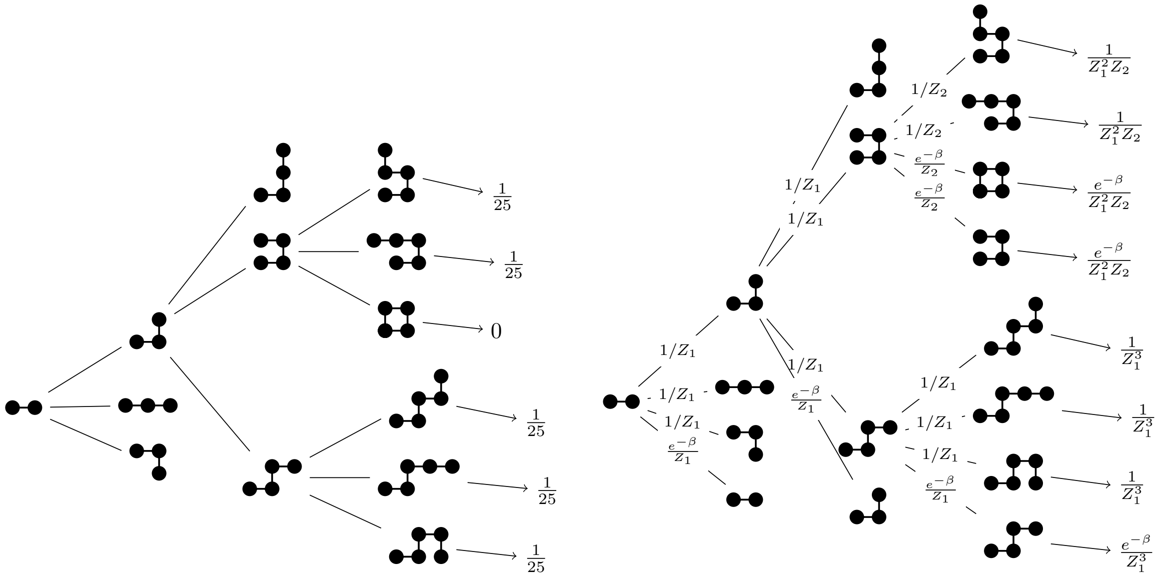


(a) $\beta = 0$ (b) $\beta = 1$ (c) $\beta = 100$

Figure 2. Examples of typical TSAW realizations at different values of the avoidance parameter β with their convex hulls. Each walk has $T = 200$ steps. Larger values of β lead to larger extended walks characterized by larger areas of their convex hulls.

$$p_i = \frac{\exp\left(-\beta n_i\right)}{\sum_{j\in\mathcal{N}} \exp\left(-\beta n_j\right)}, \tag{7}$$

where the sum over all current neighbors $\mathcal{N}$ is for normalization. The free parameter β governs the strength of the avoidance. Large values of β lead to stronger avoidance, negative values of β lead to attraction and $\beta = 0$ is the special case of the standard random walk. For a selection of β values typical examples are visualized in Fig. 2. The TSAW is not to be confused with the classical self-avoiding walk (SAW), which describes the ensemble of all configurations which do not intersect themselves each weighted the same. In Fig. 3 two partial decision trees are displayed which visualize the fundamental differences in the weights of the configurations. Even in the $\beta \to \infty$ limit ($Z_1 = 3$, $Z_2 = 2$) the weights differ. In particular its upper critical dimension is $d = 2$ [31], which means that the exponent ν, which characterizes the scaling of the end-to-end distance $r \propto T^{\nu}$, is $\nu = 1/2$ with logarithmic corrections, i.e., $r \propto T^{\nu} \left(\ln T\right)^{\alpha}$, where $\alpha = 1/4$ [33] is conjectured.



(a) SAW (b) TSAW

Figure 3. Partial decision trees for SAW and TSAW of walks up to length $T = 5$. On the right side of each tree the weight of the configuration is displayed. While the weights for the SAW are by definition uniform for every valid configuration, the TSAW not only allows self-intersection, but also has different weights depending on the history of the walk. Note that $Z_1 = 3 + \exp(-\beta)$ and $Z_2 = 2 + 2\exp(-\beta)$.

2.3. Convex Hulls

The convex hull of a set of points $\mathcal{P}$ in the plane is the smallest convex polygon enclosing every point $p \in \mathcal{P}$ and hence also every line between any pair of points. Some examples of convex hulls are visualized in Fig. 2. The construction of a convex hull of a planar point set is a solved problem, in the sense that an optimal algorithm exists [42, 43] with result-dependent run time $\mathcal{O}(T \log h)$, where T is the number of points $|\mathcal{P}|$ and h is the number of vertices of the resulting convex hull. In practice, however, suboptimal but simpler and for point sets as small as in this study ($T \approx 10^6$) faster algorithms are used. Especially for planar point sets one can exploit the fact that a polygon can be defined by the order of its vertices, instead by a list of its facets. The Graham scan [44] algorithm is based on this fact. After shifting the coordinate origin into the center of the point set, it sorts the points according to their polar coordinate. Then starting at an arbitrary point all points are filtered out which are oriented clockwise with respect to the the previous and next (not-filtered out) points. Iterating this over a full revolution, leaves only the points which constitute the vertices of the convex hull. This algorithm is dominated by the time to sort the points, which can be done in $\mathcal{O}(T \log T)$. Here, we use Andrew's monotone chain algorithm [45], which is a variation of the Graham scan sorting the points lexicographically, which is slightly faster, instead of by polar angle. Note that this type of algorithm does not generalize to 3 or higher dimensions. For those cases a different algorithm, like quickhull [46] has to be used. Before applying the exact algorithm, we reduce the size of the point set with Akl's elimination heuristic [47], which removes all points inside, in our implementation, a octagon of extreme points. Of a few tested polygons the octagon showed the best performance in the instances we typically encounter in this study.

To calculate the area A of a convex polygon, where the coordinates are sorted counterclockwise, one can sum the areas of the trapezoids extending perpendicular to the x-axis

$$A = \frac{1}{2} \sum_{i=0}^{h-1} (y_i + y_{i+1})(x_i - x_{i+1}). \tag{8}$$

The perimeter L is the sum of the line segments of consecutive points of the hull

$$L = \sum_{i=0}^{h-1} \sqrt{(x_i - x_{i+1})^2 + (y_i + y_{i+1})^2}, \tag{9}$$

with $x_h \equiv x_0$ and $y_h \equiv y_0$.

3. Results

We simulated the TSAW at two values of β. The limit case of a TSAW, which only steps on itself, if it has no other choice, was simulated at $\beta = 100$. Since the probability to step on already visited sites is exponential in β, this corresponds to the $\beta \to \infty$ case. Further, we simulated at $\beta = 1$, to capture also the case, which does sometimes voluntarily step on itself.

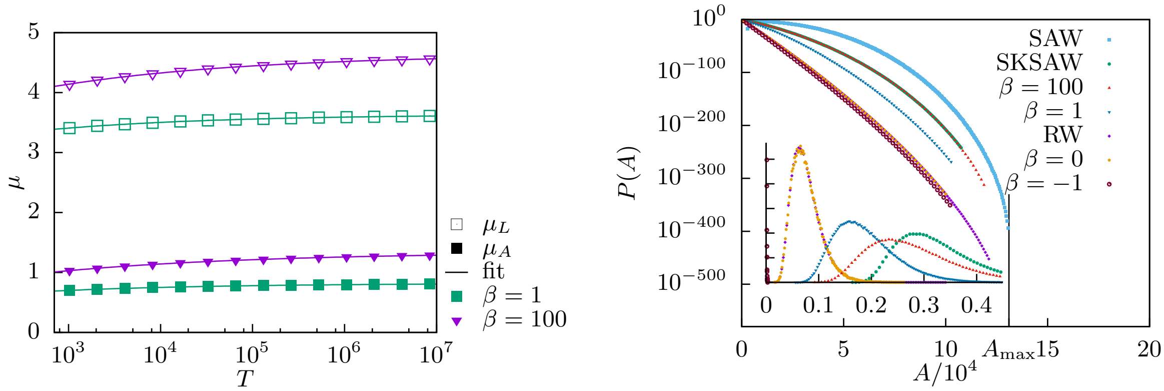
First, we will look at the behavior of the mean of the perimeter and area. Here, we used simple sampling for walk lengths in the range $T \in \{2^k | 10 \leq k \leq 23\}$. Each value is averaged over 10^6 TSAWs. Naturally, the mean of geometric volumes scale with their intrinsic dimension d_i and a typical length scale r, e.g., the end-to-end distance, as r^{d_i}. Using the scaling of r from Eq. (1), we expect the mean values of the perimeter $\langle L \rangle$ ($d_i = 1$ in $d = 2$) and the area $\langle A \rangle$ ($d_i = 2$) to scale as

$$S \propto T^{d_i \nu} \ln(T)^{d_i \alpha} \tag{10}$$

for large values of T. We can even calculate the asymptotic prefactors μ^∞ by extrapolating the scaled values for finite sizes $\mu_L = \langle L \rangle T^{-1/2} \ln(T)^{-\alpha}$ and $\mu_A = \langle A \rangle T^{-1} \ln(T)^{-2\alpha}$ to their asymptotic values μ_L^∞ and μ_A^∞. For the extrapolation, which is shown in Fig. 4(a), we use a simple power law with offset $\mu = \mu^\infty - aT^{-b}$, which were already used for this purpose in [23, 24]. The asymptotic values are listed in table 1. As expected the values for the TSAW are larger for larger β. To our knowledge, there are no analytical calculations for these asymptotic values to which we could compare to. The given error estimates are only statistical and do not include the systematic error introduced by the ad-hoc fit function. Nevertheless the convergence of the values is very well visible, confirming $\nu = 1/2$ and $\alpha = 1/4$ to be very good estimates.

Direct fits of the form Eq. (10) yield values in good agreement with the expected exponents for the end-to-end distance r at $\beta = 1$, but most other data sets lead to fits overestimating α and slightly underestimating ν. A possible, at least partial, explanation for this is be that the relation $L(r)$ is not perfectly linear for the sizes we obtained data for.

We now focus on the main result, on the distribution $P(A)$ of the convex-hull area. These results were obtained using the large-deviation Markov-chain simulations. We had to perform simulations for different "temperatures" ranges for each system size and parameter β. For example $T = 128$ at $\beta = 1$ needed seven temperatures for the right tail $\theta \in [-40, -9]$ and three more for the left $\theta \in [7, 40]$. For larger system sizes more temperatures are usually needed. For the $\beta = 1$ case at $T = 2048$ we used 32 temperatures $\theta \in [-3200, -105]$ to obtain the right



(a) Asymptotic mean values

(b) Comparison of different types of random walks

Figure 4. (a) Extrapolation of the asymptotic mean values of the perimeter and area of the convex hull. (b) Distribution of all scrutinized walk types with $T = 1024$ steps. The inset shows the peak region. Note that the standard random walk (RW) and the $\beta = 0$ TSAW coincide. Also the far right tail of the $\beta = 100$ TSAW and SKSAW coincide, but not the main region. The vertical line shows the maximum area constructable with 1024 steps, which has an area $A_{\max}$ about $512^2/2 \approx 1.3 \times 10^5$. The distributions are thus not sampled over their whole support, but a large region.

	μ_L^∞	μ_A^∞
TSAW $\beta = 0$ (exact)	3.5449...	0.7854...
TSAW $\beta = 1$	3.636(2)	0.820(1)
TSAW $\beta = 100$	4.641(3)	1.339(3)

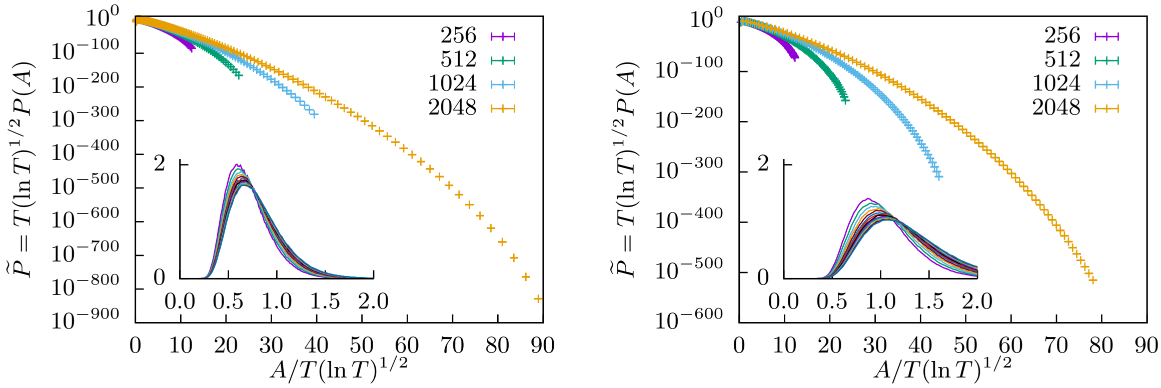
Table 1. Asymptotic mean values of the area and perimeter scaled by Eq. (10). The values are obtained by the fit shown in Fig. 4(a). The given error estimates are only statistical and do not include the systematic error introduced by the ad-hoc fit function. The exact values are from [19] and converted to a square lattice as described in [26].

tail. For the $\beta = 100$ cases we could use similar values for the temperatures. Equilibration was ensured as described in [26]. In Figure 4(b) we compare distributions of different random walk types with the result for the TSAW at different values of β. By using the large-deviation algorithm, we were able to obtain this distribution over hundreds of decades in probability, down to values as small as $P(A) \sim 10^{-800}$ for the largest value of T. Notice that while SKSAW and TSAW with high values of β show the same behavior in the far tail, where the walks are so stretched out such that trapping does not play a role anymore. In the main region however, they are clearly distinct, as is expected due to their different scaling exponent ν. Further, the parameter β can apparently be used to interpolate the tail behavior between the standard random walk case and the SKSAW case.

Since we have obtained large parts of the distribution, it would be interesting if the whole distribution scales the same as the mean values (cf. Eq. (10)). For other types of walks, the distribution of perimeter and area could indeed be scaled [23, 24, 25, 26] across their full support only knowing ν, as

$$P(S) = T^{-d\nu} \widetilde{P} \left(S T^{-d\nu} \right). \tag{11}$$

For the TSAW, this collapse, when considering the logarithmic corrections as visualized in Fig. 5, exhibits an apparent drift towards a limiting shape. Nevertheless, severe finite size effects are visible, especially in the tails but also in the main region. Despite far larger system sizes T considered, here the main region collapse is worse than for other kinds of self-avoiding walks as shown in [26]. The stronger finite size effect may be caused by the fact that all walks start on an empty lattice. This means for our case that the first steps of the walk behave differently from the last steps of the walk, when many sites are occupied. Although for the limit of large system sizes T, the latter should determine the behavior. A possible improvement to simulate TSAWs is suggested in [33], which is to simulate a much longer walk with $t \gg T$ steps and look at the last T steps.



(a) $\beta = 1$

(b) $\beta = 100$

Figure 5. Distributions of the area of the "true" self-avoiding random walk scaled according to Eq. (11) plus logarithmic corrections for different values of β and lengths T. The insets show the main region for 14 values of $T \in \{2^k | 10 \leq k \leq 23\}$ obtained by simple sampling.

The rate function Φ is defined if the distribution obeys the *large deviation principle*. This means that the distribution, for large values of T, should decay exponentially in the length T scaled by the rate function as

$$P_T(s) \approx \mathrm{e}^{-T\Phi(s)} \,. \qquad (12)$$

Usually the parameter s is between 0 and 1. We achieve this by dividing our measured area by the maximum area, i.e., by measuring $s = \frac{A}{A_{\max}}$. In two dimensions the walk configuration with maximum area is L-shaped with arms of equal length (for odd T) and therefore $A_{\max} = \frac{1}{2}\left(\frac{T+1}{2}\right)^2 \approx \frac{T^2}{8}$.

Similar to [26] we assume the rate function to be a power law

$$\Phi(s) \propto s^{\kappa}, \qquad (13)$$

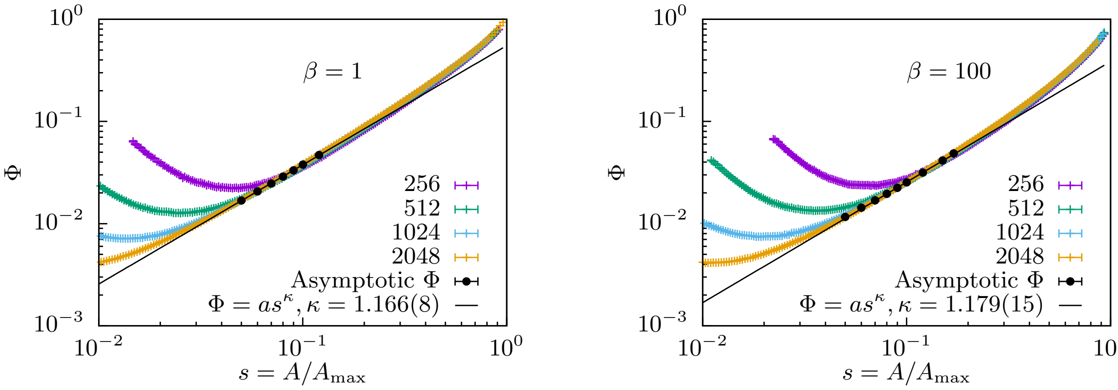
which seems to agree reasonably well with our data, since the double logarithmic plot Fig. 6 shows that the rate function appears as a straight line in the intermediate tail. The far tail is dominated by finite-size effects caused by the lattice structure, which leads to a "bending up" of the rate function. For small values of s, in the high-probability region, the rate function does not have any relevance. Assuming that the rate function is a power law Eq. (13) and scaling of the form Eq. (11) is possible, with d_i being the intrinsic dimension of the observable, e.g., $d_i = 2$

for the area, we can derive a value for the power law exponent of the rate function κ. Using the definition of the rate function Eq. (12)

$$\mathrm{e}^{-T\Phi(ST^{-d_i})} \approx T^{-d_i\nu}\widetilde{P}\left(ST^{-d_i\nu}\right) \tag{14}$$

should hold in the right tail. The $T^{-d_i\nu}$ term can be ignored next to the exponential, also the logarithmic correction is subdominant and would not allow to add any insight. Apparently the right hand side is a function of $ST^{-d_i\nu}$, such that the left hand side also has to be a function of $ST^{-d_i\nu}$. This is the case for [26]

$$\kappa = \frac{1}{d_i(1-\nu)}. \tag{15}$$



(a) (b)

Figure 6. Rate function Φ for the area with fits to the assumed power-law form. The fit is performed over a range, where the finite-size influence of the lattice should be small, but the large T behavior can be extrapolated. Finite-size effects seem more pronounced for larger values of β.

To test whether the results for the rate function in case of the TSAW obeys this relation, we estimate the value of κ from our data. Since we have data for various values of the walk length T, we first extrapolate our data point-wise to large T. For this purpose we fit a power law with offset, where the parameters depend on the value of s:

$$\Phi(s,T) = a(s)T^{b(s)} + \Phi_\infty(s). \tag{16}$$

This results in the value of interest $\Phi(s) \equiv \Phi_\infty(s)$, the parameters $a(s)$ and $b(s)$ are only auxiliary quantities. We perform this extrapolation in a region which is far away from the finite-size effects of the far tail. In this range of medium values of s the extrapolation according to Eq. (16) works robustly. Since the bins of the logarithmic histograms we use do not have the same borders for every system size T, we have a-priory not access to the same value of s for different values of T. Thus, we use cubic splines to interpolate such that we obtain results for the same value of s for all walk lengths. We found cubic splines to be sufficient since the bins are rather dense such that systematic errors introduced by the interpolation should be small. The final values $\Phi(s)$ obtained from the extrapolated values to the assumed form Eq. (13) are shown in Fig. 6 as symbols. Next, we fit power laws to this data. The values for κ obtained are within errorbars consistent with $\kappa = 7/6$ which is the expected value for the SKSAW ($\nu = 4/7$) and incompatible with the expected value $\kappa = 1$ of $\nu = 1/2$ walks. This behavior is nevertheless plausible since in the tail (large-area) region, structures, which enable self trapping, i.e., loops, are rare since they do lead to smaller areas of the convex hull than straight regions. Therefore the influence of trappings should diminish in the large area tail, which is the main difference in the behavior of SKSAW and TSAW. Without trappings the TSAW in the $\beta \to \infty$ limit is functionally identical to the SKSAW. Apparently already $\beta = 1$ is large enough to produce this behavior. Therefore it is natural that the large-area tail behaves the same as the SKSAW. On the other hand, to possibly see a range where the rate function behaves like a power law with $\kappa = 1$ according to $\nu = 1/2$, one would have to go to much larger system sizes, because one would have to obtain data to the right of the peak, but for very small values of $s = A/A_{\max} \ll 10^{-2}$, where trappings still do play a role. In particular the analysis might be hampered by the presence of the logarithmic correction to the mean end-to-end distance.

This means that the TSAW is more complex in comparison to some other types of self-avoiding walks for which it was possible to predict the tail behavior from the same exponent ν which predicts the mean behavior. The other types of random walk were under scrutiny in [26], namely the self-avoiding walk, the loop-erased random walk and the smart kinetic self-avoiding walk (SKSAW). Instead for TSAW in the large deviation region a different scaling exponent seems to hold, which is very close to the scaling exponent of the SKSAW.

4. Conclusions

We studied the behavior of the distribution of the area of the convex hull of the "true" self-avoiding walk, especially in the large deviation regime of larger than typical areas. With a sophisticated large-deviation sampling algorithm, we obtained the distribution over a large part of its support down to probabilities smaller than 10^{-800} for a typical avoidance parameter of $\beta = 1$ and a large avoidance $\beta = 100$. The distributions seem to approach a limiting scaling form when rescaled by the behavior of the mean, but with much stronger finite-size effects as compared to other types of random walks, which were previously studied.

Using this data we calculated the rate functions. The rate function seem also to behave qualitatively similar in comparison to other types of self-avoiding walks studied earlier [26] in that they seem to be well defined and well approximated by a power law. In contrast to other types of random walks, this power law can apparently not be derived from the scaling exponent of the mean values ν. Instead it seems that a second exponent governs the scaling behavior of the tail for the TSAW, which is close to $4/7$, the scaling exponent of the smart kinetic self-avoiding walk. This is plausible since the large-area region should be dominated by configurations in which no trappings are possible, which is the major difference between these types.

Finally, we also provided estimates for the relevant scale factors of the mean of area and perimeter of the convex hulls of TSAWs. They might be accessible to analytic calculations in the future.

For future numerical work it would be interesting to look for further types of random walks, which show similar effects of distinct scaling exponents for different parts of the distribution but do not show the strong logarithmic corrections to scaling, and would therefore be easier to analyze. On the other hand, it would be very exciting if one was able to obtain data in the range where the rate function exhibits the exponent $\kappa(\nu = 1/2) = 1$, with the need to simulate really large system sizes, but closer to the typical behavior.

Acknowledgments

This work was supported by the German Science Foundation (DFG) through the grant HA 3169/8-1. We also thank the GWDG (Göttingen) for providing computational resources.

References

[1] Einstein A 1906 *Annalen der Physik* **324** 371–381 ISSN 1521-3889

[2] Fama E F 1965 *Financial Analysts Journal* **21** 55–59

[3] Page L, Brin S, Motwani R and Winograd T 1999 The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66 Stanford InfoLab

[4] Newman M and Barkema G 1999 *Monte Carlo Methods in Statistical Physics* (Clarendon Press) ISBN 9780198517979

[5] Hughes B D 1996 *Random walks and random environments* (Clarendon Press Oxford)

[6] Madras N and Slade G 2013 *The Self-Avoiding Walk* (New York, NY: Springer New York) ISBN 978-1-4614-6025-1

[7] Weinrib A and Trugman S A 1985 *Phys. Rev. B* **31**(5) 2993–2997

[8] Manna S S, Dhar D and Majumdar S N 1992 *Phys. Rev. A* **46**(8) R4471–R4474

[9] Majumdar S N 1992 *Phys. Rev. Lett.* **68**(15) 2329–2331

[10] Mohr C O 1947 *American Midland Naturalist* **37** pp. 223–249

[11] Worton B J 1987 *Ecol. Model.* **38** 277–298 ISSN 0304-3800

[12] Boyle S A, Lourenco W C, da Silva L R and Smith A T 2009 *Folia Primatol.* **80** 33–42

[13] Dumonteil E, Majumdar S N, Rosso A and Zoia A 2013 *Proceedings of the National Academy of Sciences* **110** 4239–4244

[14] Lanoiselée Y and Grebenkov D S 2017 *Phys. Rev. E* **96**(2) 022144

[15] Letac G and Takács L 1980 *The American Mathematical Monthly* **87** 142–142 ISSN 00029890, 19300972

[16] Letac G 1993 *Journal of Theoretical Probability* **6** 385–387 ISSN 1572-9230

[17] Randon-Furling J, Majumdar S N and Comtet A 2009 *Phys. Rev. Lett.* **103**(14) 140602

[18] Majumdar S N, Comtet A and Randon-Furling J 2010 *Journal of Statistical Physics* **138** 955–1009 ISSN 1572-9613

[19] Eldan R 2014 *Electron. J. Probab.* **19** no. 45, 1–34 ISSN 1083-6489

[20] Grebenkov D S, Lanoiselée Y and Majumdar S N 2017 *Journal of Statistical Mechanics: Theory and Experiment* **2017** 103203

[21] Kabluchko Z and Zaporozhets D 2016 *Transactions of the American Mathematical Society* **368** 8873–8899

[22] Goldman A 1996 *Probability Theory and Related Fields* **105** 57–83 ISSN 1432-2064

[23] Claussen G, Hartmann A K and Majumdar S N 2015 *Phys. Rev. E* **91**(5) 052104

[24] Dewenter T, Claussen G, Hartmann A K and Majumdar S N 2016 *Phys. Rev. E* **94**(5) 052120

[25] Schawe H, Hartmann A K and Majumdar S N 2017 *Phys. Rev. E* **96**(6) 062101

[26] Schawe H, Hartmann A K and Majumdar S N 2018 *Phys. Rev. E* **97**(6) 062159

[27] Cardy J 2005 *Annals of Physics* **318** 81 – 118 ISSN 0003-4916 special Issue

[28] Lawler G F, Schramm O and Werner W 2002 *arXiv preprint math/0204277*

[29] Lawler G F, Schramm O and Werner W 2011 *Conformal Invariance Of Planar Loop-Erased Random Walks and Uniform Spanning Trees* (New York, NY: Springer New York) pp 931–987 ISBN 978-1-4419-9675-6

[30] Kennedy T 2015 *Journal of Statistical Physics* **160** 302–320 ISSN 1572-9613

[31] Amit D J, Parisi G and Peliti L 1983 *Phys. Rev. B* **27**(3) 1635–1645

[32] Pietronero L 1983 *Phys. Rev. B* **27**(9) 5887–5889

[33] Grassberger P 2017 *Phys. Rev. Lett.* **119**(14) 140601

[34] Metropolis N, Rosenbluth A W, Rosenbluth M N, Teller A H and Teller E 1953 *The Journal of Chemical Physics* **21** 1087–1092

[35] Hartmann A K 2014 *Phys. Rev. E* **89**(5) 052103

[36] Hartmann A K 2002 *Phys. Rev. E* **65**(5) 056102

[37] Wolfsheimer S, Burghardt B and Hartmann A K 2007 *Algorithms for Molecular Biology* **2** 9 ISSN 1748-7188

[38] Fieth P and Hartmann A K 2016 *Phys. Rev. E* **94**(2) 022127

[39] Engel A, Monasson R and Hartmann A K 2004 *Journal of Statistical Physics* **117** 387–426 ISSN 1572-9613

[40] Hartmann A K 2011 *The European Physical Journal B* **84** 627–634 ISSN 1434-6036

[41] Hartmann A K and Mézard M 2018 *Phys. Rev. E* **97**(3) 032128

[42] Kirkpatrick D and Seidel R 1986 *SIAM Journal on Computing* **15** 287–299

[43] Chan T M 1996 *Discrete & Computational Geometry* **16** 361–368 ISSN 1432-0444

[44] Graham R 1972 *Information Processing Letters* **1** 132–133

[45] Andrew A 1979 *Information Processing Letters* **9** 216 – 219 ISSN 0020-0190

[46] Barber C B, Dobkin D P and Huhdanpaa H 1996 *ACM Trans. Math. Softw.* **22** 469–483 URL `http://www.qhull.org`

[47] Akl S G and Toussaint G T 1978 *Information Processing Letters* **7** 219 – 222 ISSN 0020-0190

A.4. Ground state energy of noninteracting fermions with a random energy spectrum

Hendrik Schawe is the author of the thesis at hand. Alexander K. Hartmann is the supervising professor of H. Schawe. Satya N. Majumdar is directeur de recherche at the Laboratoire de Physique Théorique et Modèles Statistiques (LPTMS) at the Université Paris–Sud in Orsay, France. Grégory Schehr is permanent CNRS research scientist also at the LPTMS in Orsay, France.

The topic of this article was conceived by G. Schehr and S. N. Majumdar. During a stay at the LPTMS S. N. Majumdar suggested A. K. Hartmann and H. Schawe to do simulations of a problem they looked at for some time. At this stage the problem under scrutiny was the sum of the K largest values of N i.i.d. random numbers. H. Schawe completed simulations, for which to be feasible some improvements to the used importance sampling algorithm were needed, which were developed in discussions between A. K. Hartmann and H. Schawe. While it turned out that the sum of the largest entries is an already solved problem, G. Schehr and S. N. Majumdar conceived a very closely related problem: the sum of the K smallest values of N i.i.d. random numbers, which at the same time can be interpreted as a toy model in the spirit of Derrida's *random-energy model*. With the before developed algorithmic improvements H. Schawe gathered large parts of the distributions, which confirm the analytical results obtained by G. Schehr and S. N. Majumdar. The first draft was prepared by G. Schehr, H. Schawe extended it with the numerical results and methods. The final draft contains improvements originating from all four authors.

A Letters Journal Exploring the Frontiers of Physics

November 2018

EPL, **124** (2018) 40005
doi: 10.1209/0295-5075/124/40005

www.epljournal.org

Ground-state energy of noninteracting fermions with a random energy spectrum

Hendrik Schawe[1], Alexander K. Hartmann[1], Satya N. Majumdar[2] and Grégory Schehr[2]

[1] *Institut für Physik, Universität Oldenburg - D-26111 Oldenburg, Germany*
[2] *Univ. Paris-Sud, CNRS, LPTMS, UMR 8626 - Orsay F-91405, France*

received 29 August 2018; accepted in final form 9 November 2018
published online 11 December 2018

PACS `05.10.Ln` – Monte Carlo methods
PACS `75.10.Nr` – Spin-glass and other random models
PACS `05.20.-y` – Classical statistical mechanics

Abstract – We derive analytically the full distribution of the ground-state energy of K noninteracting fermions in a disordered environment, modelled by a Hamiltonian whose spectrum consists of N i.i.d. random energy levels with distribution $p(\varepsilon)$ (with $\varepsilon \geq 0$), in the same spirit as the "Random Energy Model". We show that for each fixed K, the distribution $P_{K,N}(E_0)$ of the ground-state energy E_0 has a universal scaling form in the limit of large N. We compute this universal scaling function and show that it depends only on K and the exponent α characterizing the small ε behaviour of $p(\varepsilon) \sim \varepsilon^\alpha$. We compared the analytical predictions with results from numerical simulations. For this purpose we employed a sophisticated importance-sampling algorithm that allowed us to obtain the distributions over a large range of the support down to probabilities as small as 10^{-160}. We found asymptotically a very good agreement between analytical predictions and numerical results.

Copyright © EPLA, 2018

The celebrated "Random Energy Model" (REM) of Derrida [1] has continued to play a central role in understanding different aspects of classical disordered systems, including spin-glasses, directed polymers in random media and many other systems. In the REM, one typically has N energy levels which are considered to be independent and identically distributed (i.i.d.) random variables, each drawn from a probability distribution function (PDF) $p(\varepsilon)$. Typical observables of interest are the partition function, free energy, etc. The REM can also be useful as a toy model in quantum disordered systems. For example, let us consider a single quantum particle in a disordered medium with the Hamiltonian $\hat{h}$. We will assume that the spectrum of the operator $\hat{h}$ has a finite number of states N (for instance a quantum particle on a lattice of finite size and a random onsite potential, as in the Anderson model). In general, solving exactly the spectrum of such an operator is hard, for a generic random potential. One possible approximation, in the spirit of the REM in classical disordered systems, would be to consider the toy model where one replaces the spectrum of the actual Hamiltonian by N *ordered* i.i.d. energy levels $\varepsilon_1 \leq \varepsilon_2 \leq \cdots \leq \varepsilon_N$ each drawn from the common PDF $p(\varepsilon)$. Without loss of generality, we will also assume that the Hamiltonian $\hat{h}$ has only positive eigenvalues. This would mean that, in the corresponding toy model, the PDF $p(\varepsilon)$ is supported on $[0, +\infty)$. It is well known that, in a strongly disordered quantum system, where all single-particle eigenfunctions are localised in space, the energy levels can be approximated by i.i.d. random variables (see, *e.g.*, [2]). Therefore, the REM that we consider here will be relevant in such strongly localised part of the spectrum of a disordered Hamiltonian.

Now consider a system of K noninteracting fermions with the Hamiltonian $\hat{H}_K = \sum_{i=1}^K \hat{h}_i$ where $\hat{h}_i$ is the single-particle Hamiltonian associated with the i-th particle. The ground state of this many-body system would correspond to filling up the single-particle spectrum up to the Fermi level ε_K, with one particle occupying each of the states with energies $\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_K$. The ground-state energy E_0 of this many-body system is therefore given by

$$E_0 = \sum_{i=1}^{K} \varepsilon_i. \tag{1}$$

Clearly, E_0 is a random variable, which fluctuates from one realisation of the disorder to another. Given $p(\varepsilon)$, we are interested in computing the distribution $P_{K,N}$ of E_0, for fixed K (*i.e.*, the number of fermions) and N (*i.e.*, the number of levels). We note that, for $K = 1$, $E_0 = \varepsilon_1$ is just the minimum of a set of N i.i.d. random variables and is described by the well-known extreme value statistics [3]. Thus, for general value of K, in particular, it would be interesting to know how sensitive the distribution of E_0 is to the choice of $p(\varepsilon)$. For instance, is there any universal feature of the distribution of E_0 that is independent of $p(\varepsilon)$? We note that E_0 is a sum of random variables, but these random variables are not independent due to the ordering $\varepsilon_1 \leq \varepsilon_2 \leq \cdots \leq \varepsilon_N$ (even though the original unordered random variables are independent). Had they been independent, the sum E_0 in eq. (1), by virtue of the Central Limit Theorem, would converge to a shifted and scaled Gaussian random variable. Here, this is not the case, as the ordering induces nontrivial correlations between these variables. The fact that the ordering introduces correlations between otherwise i.i.d. random variables was observed by Rényi in the context of positive i.i.d. random variables, each distributed purely exponentially [4].

In this paper, we compute exactly the PDF $P_{K,N}(E_0)$ for arbitrary K, N and $p(\varepsilon)$ and show that, indeed, a universal feature emerges in the large-N limit. It turns out that the limiting distribution of E_0, for large N, depends only on the small ε behaviour of $p(\varepsilon) \approx B\,\varepsilon^{\alpha}$, with $\alpha > -1$, but is otherwise independent of the rest of the features of $p(\varepsilon)$. For fixed α and fixed K, as $N \to \infty$, we show that the distribution of the ground-state energy converges to a limiting scaling form

$$P_{K,N}(E_0) \approx b\,N^{\frac{1}{\alpha+1}} F_K^{(\alpha)}\left(b\,N^{\frac{1}{\alpha+1}}\,E_0\right), \tag{2}$$

where $b = (B/(\alpha+1))^{1/(\alpha+1)}$ is just a scale factor. The scaling function $F_K^{(\alpha)}(z)$ (with $z \in [0,+\infty)$) is universal and depends only on α and K. We show that the Laplace transform of $F_K^{(\alpha)}(z)$ is given explicitly by

$$\begin{aligned}\int_0^\infty F_K^{(\alpha)}(z)\mathrm{e}^{-\lambda\,z}\,\mathrm{d}z &= \frac{(\alpha+1)^K}{\Gamma(K)\lambda^{(\alpha+1)(K-1)}}\\ &\times \int_0^\infty x^{\alpha}\mathrm{e}^{-\lambda\,x - x^{\alpha+1}}\left[\gamma(\alpha+1,\lambda\,x)\right]^{K-1}\,\mathrm{d}x\;,\end{aligned} \tag{3}$$

where $\gamma(a,x) = \int_0^x \mathrm{d}u\,u^{a-1}\mathrm{e}^{-u}$ is the incomplete gamma function. While we can invert formally this Laplace transform (3), it does not have a simple expression for generic α. However, we can derive the asymptotic behaviour of $F_K^{(\alpha)}(z)$

$$F_K^{(\alpha)}(z) \approx \begin{cases} c_1\,z^{(\alpha+1)K-1}, & z \to 0,\\ c_2\,z^{\alpha}\,\exp\left[-\left(\dfrac{z}{K}\right)^{\alpha+1}\right], & z \to \infty, \end{cases} \tag{4}$$

where $c_1 = \frac{[\Gamma(\alpha+2)]^K}{\Gamma(K+1)\Gamma((\alpha+1)K)}$ and $c_2 = \frac{(\alpha+1)K^{K-\alpha-2}}{\Gamma(K)}$ are constants. For the extreme-value case $K = 1$ our result, $F_1^{(\alpha)}(z) = (\alpha+1)\,z^{\alpha}\,\mathrm{e}^{-z^{\alpha+1}}$, coincides with the well-known Weibull scaling function [3]. Note that here we are interested in the sum of K lowest i.i.d. variables supported over $[0,+\infty)$. We remark that in the statistics literature, in a completely different context, the sum of the top K values of a set of i.i.d. random variables with an unbounded support has been studied [5,6]. However, we have not found our results (2) and (3) in the statistics literature.

We start with a set of N positive i.i.d. random variables $\{x_1, x_2, \cdots, x_N\}$, each drawn from a common distribution $p(x)$, supported on $[0,+\infty)$. The joint distribution of these variables is simply $P(x_1,\cdots,x_N) = \prod_{i=1}^N p(x_i)$. At this stage, these variables are unordered. We are interested in the first K ordered variables $\{\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_K\}$ with $K \leq N$. This ordering makes these K variables correlated. Indeed, the joint distribution of the K lowest ordered variables can be written explicitly as

$$\begin{aligned}P(\varepsilon_1,\cdots,\varepsilon_K) &= \frac{\Gamma(N+1)}{\Gamma(N-K+1)}\\ &\times \prod_{i=1}^K p(\varepsilon_i)\prod_{i=2}^K \Theta(\varepsilon_i - \varepsilon_{i-1})\left[\int_{\varepsilon_K}^\infty p(u)\,\mathrm{d}u\right]^{N-K}.\end{aligned} \tag{5}$$

This result can be easily understood as follows. We first choose the K distinct variables from an i.i.d. set of N variables. The number of ways this can be done is simply the combinatorial factor $N(N-1)\cdots(N-K+1) = \Gamma(N+1)/\Gamma(N-K+1)$ in eq. (5). The probability that they are ordered is $\prod_{i=1}^K p(\varepsilon_i)\prod_{i=2}^K \Theta(\varepsilon_i - \varepsilon_{i-1})$, where the Heaviside theta functions ensure the ordering. In addition, we have to ensure that the $N-K$ remaining variables are bigger than ε_K, *i.e.*, the largest value among the first K ordered variables. Since these $N-K$ variables are i.i.d., this gives the last factor in eq. (5). The formula in eq. (5) is exact for any $p(\varepsilon)$, K and N. Given the joint PDF (5), we are interested in the distribution $P_{K,N}(E_0)$ of the ground-state energy E_0 in eq. (1). We therefore have

$$P_{K,N}(E_0) = \int P(\varepsilon_1,\cdots,\varepsilon_K)\delta\left(E_0 - \sum_{i=1}^K \varepsilon_i\right)\prod_{i=1}^K \mathrm{d}\varepsilon_i. \tag{6}$$

The form of this equation naturally suggests to consider the Laplace transform with respect to (w.r.t.) E_0

$$\langle \mathrm{e}^{-sE_0}\rangle = \int_0^\infty P_{K,N}(E_0)\,\mathrm{e}^{-sE_0}\,\mathrm{d}E_0. \tag{7}$$

Taking the Laplace transform of eq. (6) gives

$$\begin{aligned}\langle \mathrm{e}^{-sE_0}\rangle &= \frac{\Gamma(N+1)}{\Gamma(N-K+1)}\int_0^\infty \mathrm{d}\varepsilon_K\,p(\varepsilon_K)\mathrm{e}^{-s\,\varepsilon_K}\\ &\times \left[\int_{\varepsilon_K}^\infty p(u)\mathrm{d}u\right]^{N-K} J_{K-1}(\varepsilon_K),\end{aligned} \tag{8}$$

where

$$J_{K-1}(\varepsilon_K) = \int \prod_{i=1}^{K-1} p(\varepsilon_i)\mathrm{e}^{-s\varepsilon_i}\,\mathrm{d}\varepsilon_i \prod_{i=2}^K \Theta(\varepsilon_i - \varepsilon_{i-1}). \tag{9}$$

This multiple integral (9) has a nested structure and can be evaluated easily by induction and one gets

$$J_{K-1}(\varepsilon_K) = \frac{1}{(K-1)!}\left[\int_0^{\varepsilon_K} \mathrm{d}u\,\mathrm{e}^{-su}p(u)\right]^{K-1}. \tag{10}$$

Using this result in eq. (8), and also replacing, for later convenience, $\int_y^\infty \mathrm{d}u\, p(u) = 1 - \int_0^y \mathrm{d}u\, p(u)$, we get the exact formula

$$\langle \mathrm{e}^{-sE_0}\rangle = K\binom{N}{K}\int_0^\infty \mathrm{d}y\, p(y)\,\mathrm{e}^{-sy}\left[1-\int_0^y \mathrm{d}u\, p(u)\right]^{N-K} \times \left[\int_0^y \mathrm{d}v\, p(v)\,\mathrm{e}^{-sv}\right]^{K-1}. \tag{11}$$

This formula has a simple interpretation. Taking the Laplace transform is equivalent to breaking the system into two species of random variables of size K and $N-K$ (this can be done in $\binom{N}{K}$ ways): Each member of the first species of size K comes with an effective weight $p(\varepsilon)\,\mathrm{e}^{-s\varepsilon}$, while in the second species of size $N-K$ each member comes with an effective weight $p(\varepsilon)$. We first fix the K-th variable to have a value y, whose weight is $p(y)\,\mathrm{e}^{-sy}$. The members of the second species should each be bigger than y (explaining the factor $[\int_y^\infty p(u)\,\mathrm{d}u]^{N-K}$), while the rest of the $(K-1)$ members of the first species should each be smaller than y, explaining the factor $[\int_0^y \mathrm{d}v\,\mathrm{e}^{-sv}p(v)]^{K-1}$. Finally, the biggest variable among the members of the first species can be any of the K members, explaining the factor K multiplying the binomial coefficient $\binom{N}{K}$ in eq. (11). With this interpretation, it is clear that eq. (11) can be easily generalized to any linear statistics of the form $L_K = \sum_{i=1}^K f(\varepsilon_i)$, where $f(\varepsilon)$ is an arbitrary function. The ground-state energy E_0 considered here corresponds to choosing $f(\varepsilon)=\varepsilon$. For general $f(\varepsilon)$ the effective weight of each member of the first species discussed above is just $p(\varepsilon)\,\mathrm{e}^{-sf(\varepsilon)}$. Hence the formula in eq. (11) generalises to

$$\langle \mathrm{e}^{-sL_K}\rangle = K\binom{N}{K}\int_0^\infty \mathrm{d}y\, p(y)\,\mathrm{e}^{-sf(y)}\left[\int_y^\infty p(u)\,\mathrm{d}u\right]^{N-K} \times \left[\int_0^y \mathrm{d}v\, p(v)\,\mathrm{e}^{-sf(v)}\right]^{K-1}. \tag{12}$$

In this paper, we will focus only on the case $f(\varepsilon)=\varepsilon$. Below, we thus start with the exact result in eq. (11) and analyse its behaviour in the large-N limit.

To understand the large-N scaling limit, it is instructive to start with the $K=1$ case. In this case, $E_0=\varepsilon_1$ is just the minimum of a set of N i.i.d. random variables, each drawn from $p(\varepsilon)$. In this case, eq. (11) reads (upon setting $K=1$)

$$\langle \mathrm{e}^{-sE_0}\rangle = N\int_0^\infty \mathrm{d}y\, p(y)\,\mathrm{e}^{-sy}\left[1-\int_0^y \mathrm{d}u\, p(u)\right]^{N-1}, \tag{13}$$

where we replaced $\int_y^\infty \mathrm{d}u\, p(u) = 1-\int_0^y \mathrm{d}u\, p(u)$, using the normalisation of $p(u)$. In the large-N limit, the dominant contribution to the integral over y comes from the regime of y where the integral $\int_0^y \mathrm{d}u\, p(u)$ is of order $O(1/N)$. For other values of y, the contribution is exponentially small in N, for large N. Hence, we see that, in the large-N limit, only the small-y behaviour of $p(y)$ matters. Let

$$p(y) \underset{y\to 0}{\approx} B\, y^\alpha, \tag{14}$$

where $\alpha > -1$ in order that $p(y)$ is normalisable and clearly $B>0$. Substituting this leading-order behaviour of $p(y)$ for small y (14) in eq. (13), we get

$$\langle \mathrm{e}^{-sE_0}\rangle \approx B\,N\int_0^\infty \mathrm{d}y\, y^\alpha\,\mathrm{e}^{-sy}\exp\left(-\frac{B\,N}{\alpha+1}\,y^{\alpha+1}\right). \tag{15}$$

Performing the change of variable $y=\left(\frac{\alpha+1}{B\,N}\right)^{\frac{1}{\alpha+1}}x$, we get

$$\langle \mathrm{e}^{-sE_0}\rangle \approx (\alpha+1)\int_0^\infty \mathrm{d}x\, x^\alpha \exp\left(-\frac{s}{b\,N^{\frac{1}{\alpha+1}}}\,x - x^{\alpha+1}\right), \tag{16}$$

where $b=(B/(\alpha+1))^{1/(\alpha+1)}$. Inverting the Laplace transform formally, we obtain the scaling form given in eq. (2) with $K=1$ and the scaling function $F_1^{(\alpha)}(z)$ has its Laplace transform as in (3) with $K=1$. Inverting this Laplace transform exactly, we recover the Weibull scaling function $F_1^{(\alpha)}(z)=(\alpha+1)z^\alpha\,\mathrm{e}^{-z^{\alpha+1}}$. The calculation for $K=1$ shows that only the small-y behaviour of $p(y)$ matters in the limit of large N. Furthermore, for $K=1$, we see that the typical value of E_0 scales as $N^{-\frac{1}{\alpha+1}}$ for large N. We then anticipate that, even for $K>1$, the typical scale of E_0 will remain the same $E_0\sim N^{-\frac{1}{\alpha+1}}$ for large N. Below, we indeed use this typical scale for E_0 (and verify *a posteriori*) and compute the scaling function $F_K^{(\alpha)}(z)$ for general K in eq. (2).

We now derive the main results in eqs. (2) and (3) for all $K\geq 1$. Anticipating the scaling $E_0\sim N^{-\frac{1}{\alpha+1}}$ as mentioned above, we set

$$E_0 = \frac{1}{b}\,N^{-\frac{1}{\alpha+1}}\,z, \tag{17}$$

where b is a constant to be fixed later and the scaled ground-state energy z is of order $O(1)$. Substituting this scaling form (17) in eq. (11), we see that the left-hand side (l.h.s.) reads $\langle \mathrm{e}^{-sE_0}\rangle = \langle \mathrm{e}^{-s\,N^{-\frac{1}{\alpha+1}}\,z/b}\rangle = \langle \mathrm{e}^{-\lambda\,z}\rangle$, where $\lambda = N^{-\frac{1}{\alpha+1}}\,s/b$ is the rescaled Laplace variable. We will take the $N\to\infty$ limit, keeping λ fixed. This then corresponds to $s\to\infty$ limit. On the right-hand side (r.h.s.) of eq. (11) we make a change of variable $s\,y=\tilde{x}$ as well as $u=\tilde{u}/s$ and $v=\tilde{v}/s$. This gives

$$\langle \mathrm{e}^{-\lambda z}\rangle = \frac{K}{s^K}\binom{N}{K}\int_0^\infty \mathrm{d}\tilde{x}\, p\left(\frac{\tilde{x}}{s}\right)\mathrm{e}^{-\tilde{x}} \times \left(1-\frac{1}{s}\int_0^{\tilde{x}}\mathrm{d}\tilde{u}\, p\left(\frac{\tilde{u}}{s}\right)\right)^{N-K}\left(\int_0^{\tilde{x}}\mathrm{d}\tilde{v}\, p\left(\frac{\tilde{v}}{s}\right)\mathrm{e}^{-\tilde{v}}\right)^{K-1}. \tag{18}$$

In the large-s limit, we use $p(y)\approx B\,y^\alpha$ to leading order. Inserting this behaviour in eq. (18), we get

$$\langle \mathrm{e}^{-\lambda z}\rangle \approx \frac{K\,B^K}{s^{(\alpha+1)K}}\binom{N}{K}\int_0^\infty \mathrm{d}\tilde{x}\,\tilde{x}^\alpha\,\mathrm{e}^{-\tilde{x}} \times \left(\mathrm{e}^{-\frac{B(N-K)}{(\alpha+1)\,s^{\alpha+1}}\,\tilde{x}^{\alpha+1}}\right)[\gamma(\alpha,\tilde{x})]^{K-1}, \tag{19}$$

where we recall that $\gamma(a,x)=\int_0^x \mathrm{d}u\, u^{a-1}\mathrm{e}^{-u}$ is the incomplete gamma function. We now use $s=(\lambda\, b)\, N^{\frac{1}{\alpha+1}}$ and choose

$$b=\left(\frac{B}{\alpha+1}\right)^{\frac{1}{\alpha+1}}. \tag{20}$$

Furthermore, in the large-N limit $K\binom{N}{K}\sim N^K/\Gamma(K)$. Using these results, and rescaling $\tilde{x}=\lambda\, x$, we arrive at

$$\begin{aligned}\langle \mathrm{e}^{-\lambda\, z}\rangle =\; & \frac{(\alpha+1)^K}{\Gamma(K)\lambda^{(\alpha+1)(K-1)}} \\ & \times \int_0^\infty x^{\alpha}\mathrm{e}^{-\lambda x - x^{\alpha+1}}\left[\gamma(\alpha+1,\lambda\, x)\right]^{K-1}\,\mathrm{d}x.\end{aligned} \tag{21}$$

This clearly shows that the distribution of the rescaled random variable $z=(E_0\, b)\, N^{\frac{1}{\alpha+1}}$ (see eq. (17)) converges to an N-independent form $F_k^{(\alpha)}(z)$ for large N, whose Laplace transform is given by $\int_0^\infty F_K^{(\alpha)}(z)\mathrm{e}^{-\lambda\, z}\,\mathrm{d}z=\langle \mathrm{e}^{-\lambda\, z}\rangle$. Therefore, eq. (21) demonstrates the result announced in eq. (3).

Special cases $\alpha=0$. – In this case eq. (3), using $\gamma(1,\lambda\, x)=1-\mathrm{e}^{-\lambda x}$, reduces to

$$\begin{aligned}&\int_0^\infty F_K^{(0)}(z)\mathrm{e}^{-\lambda\, z}\,\mathrm{d}z = \\ &\frac{1}{\Gamma(K)\lambda^{K-1}}\int_0^\infty \mathrm{d}x\, \mathrm{e}^{-(\lambda+1)x}\left(1-\mathrm{e}^{-\lambda\, x}\right)^{K-1} = \\ &\frac{\Gamma(1+1/\lambda)}{\lambda^k\,\Gamma(k+1+1/\lambda)}.\end{aligned} \tag{22}$$

Using the properties of the Γ-function, one can express the r.h.s. of (22) as a simple product

$$\int_0^\infty F_K^{(0)}(z)\mathrm{e}^{-\lambda\, z}\,\mathrm{d}z=\prod_{m=1}^{K}\frac{1}{1+m\,\lambda}. \tag{23}$$

To invert this Laplace transform, we note that the r.h.s. has simple poles at $\lambda=-1/m$ with $m=0,1,\cdots,K$. Evaluating carefully the residues at these poles, we can invert this Laplace transform explicitly and get

$$F_K^{(0)}(z)=\sum_{n=1}^{K}(-1)^{K-n}\frac{n^{K-1}}{(K-n)!\,n!}\,\mathrm{e}^{-z/n}. \tag{24}$$

For instance,

$$F_1^{(0)}(z) = \mathrm{e}^{-z}, \tag{25}$$

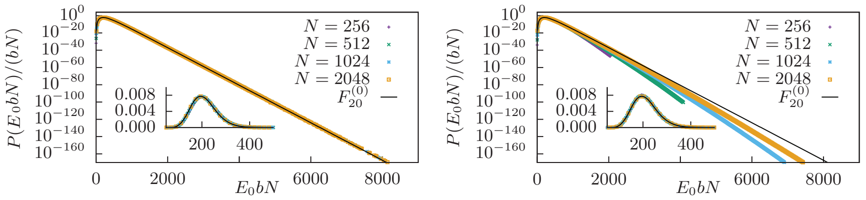
$$F_2^{(0)}(z) = \mathrm{e}^{-z/2}-\mathrm{e}^{-z}, \tag{26}$$

$$F_3^{(0)}(z) = \frac{3}{2}\mathrm{e}^{-z/3}-2\,\mathrm{e}^{-z/2}+\frac{1}{2}\,\mathrm{e}^{-z}. \tag{27}$$

Numerical simulations. – Next, we verify our analytical predictions via numerical simulations. To test the prediction of the scaling behaviour in eq. (2), as well as to test the universality of the associated scaling function $F_K^{(\alpha)}(z)$, we consider four different distributions for the energy levels: a) an exponential distribution $p(\varepsilon)=\mathrm{e}^{-\varepsilon}\,\Theta(\varepsilon)$, b) an half-Gaussian distribution $p(\varepsilon)=\sqrt{\frac{2}{\pi}}\,\mathrm{e}^{-\varepsilon^2}\,\Theta(\varepsilon)$, c) a Pareto distribution $p(\varepsilon)=\frac{2}{\varepsilon^3}\Theta(\varepsilon-1)$ and d) $p(\varepsilon)=\varepsilon\mathrm{e}^{-\varepsilon}\,\Theta(\varepsilon)$. The cases a) and b) clearly correspond to $\alpha=0$. Hence we expect the scaling function to be given by $F_K^{(0)}(z)$ in eq. (24). The Pareto case c), with support over $[1,+\infty)$, also corresponds to the $\alpha=0$ case, as seen easily after a trivial shift $\varepsilon\to\varepsilon-1$. Hence, in this case as well, we expect the scaling function to be given by $F_K^{(0)}(z)$. However, case d) is different as it corresponds to $\alpha=1$ and hence the scaling function should be given by $F_K^{(1)}(z)$. In fig. 1, we compare the simulation results with the analytical predictions and find very good agreement. Note that in cases a)–c), the scaling function $F_K^{(0)}(z)$ has an explicit expression as in eq. (24). Hence, it is easy to compare directly the simulation results with this expression (as in fig. 1(a)–(c)). However, for case d), where $\alpha=1$, we do not have a simple explicit formula for $F_K^{(1)}(z)$, though we have explicitly given its Laplace transform in eq. (3) with $\alpha=1$. Hence, to compare with the simulation results, we first needed to invert this Laplace transform using an arbitrary precision library [7]. This comparison is shown in fig. 1(d).

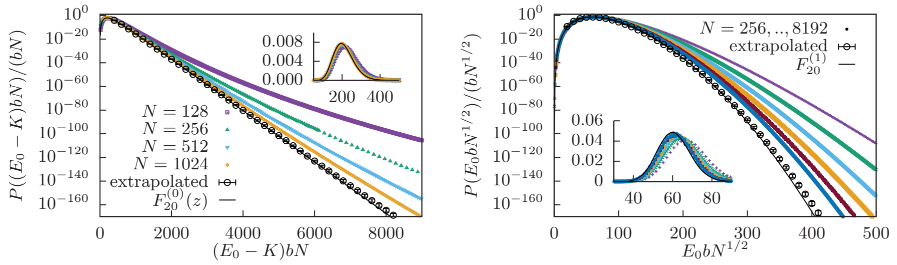
To obtain the presented numerical results one has to generate N random numbers according to the desired probability density $p(\varepsilon)$. Using a standard method, we first choose a uniform random number $\eta\in[0,1]$ and then generate ε using the formula, $\int_0^\varepsilon p(\varepsilon')\,\mathrm{d}\varepsilon'=\eta$. The exponential a) and Pareto c) cases can be trivially obtained using this relation [8]. In the half-Gaussian case b), the Gaussian random numbers can be generated using the Box-Muller method [8]. In the case d), $p(\varepsilon)=\varepsilon\mathrm{e}^{-\varepsilon}$, the above relation reads $\eta=\int_0^\varepsilon p(\varepsilon')\mathrm{d}\varepsilon'=1-(1+\varepsilon)\mathrm{e}^{-\varepsilon}$, which can also be inverted using the -1 branch of the Lambert W function [9] $\varepsilon=-W_{-1}\left(\frac{\eta-1}{e}\right)-1$. To evaluate the Lambert W function, we use the GSL implementation [10].

The sum E_0 in eq. (1) is completely determined by the values $\boldsymbol{\eta}=(\eta_1,\dots,\eta_N)$. If one simply generates many times vectors $\boldsymbol{\eta}$ of independent uniform random numbers and correspondingly obtained random numbers $(\varepsilon_1,\dots,\varepsilon_N)$, one will obtain only typical results for E_0, *i.e.*, those having a high enough probability. Here, we sample the distributions over a broad range of the support, also in the far tails, where the probabilities are extremely small. For this purpose, we use a well-tested importance sampling scheme [11,12]. Here the vectors $\boldsymbol{\eta}$ are sampled using the Metropolis algorithm including a bias of samples away from the main part of the distribution. We use a bias $\mathrm{e}^{-E_0/T}$, where T is a "temperature" parameter which can be positive and negative and allows us to address different ranges of the distribution. Since the bias is known, the Metropolis results can be corrected for the bias to obtain the actual distribution. This enables us to gather good statistics also in the far tails.



(a) $p(\varepsilon) = e^{-\varepsilon}\Theta(\varepsilon)$

(b) $p(\varepsilon) = \sqrt{\frac{2}{\pi}}e^{-\varepsilon^2}\Theta(\epsilon)$



(c) $p(\varepsilon) = \frac{2}{\varepsilon^3}\Theta(\varepsilon - 1)$

(d) $p(\varepsilon) = \varepsilon e^{-\varepsilon}\Theta(\epsilon)$

Fig. 1: (Colour online) Scaled distribution $P_{K,N}(E_0)$ for $K = 20$, for different values of N and for four different distribution $p(\varepsilon)$. The insets show the behaviour near the peaks for the four different cases. Panel (a) shows exponentially distributed $p(\varepsilon) = \mathrm{e}^{-\varepsilon}\,\Theta(\varepsilon)$ which corresponds to $\alpha = 0$ and $b = 1$ (see eq. (2)). Panel (b) shows half-Gaussian distributed $p(\varepsilon) = \frac{\sqrt{2}}{\sqrt{\pi}}\mathrm{e}^{-\varepsilon^2/2}\,\Theta(\varepsilon)$ corresponding to $\alpha = 0$ and $b = p(0) = \frac{2}{\sqrt{2\pi}}$. Panel (c) shows Pareto distributed energy levels $p(\varepsilon) = \frac{2}{\varepsilon^3}\Theta(\varepsilon - 1)$. After shifting $\varepsilon \to \varepsilon - 1$, *i.e.*, $E_0 \to E_0 - K$, this falls in the $\alpha = 0$ universality, with $b = 2$. The finite-size extrapolation is shown in black circles (see text and eq. (29) with $\beta = 1$). Panel (d) shows energy levels distributed according to $p(\varepsilon) = \varepsilon\mathrm{e}^{-\varepsilon}\,\Theta(\varepsilon)$. This corresponds to the $\alpha = 1$ universality class, with the scaling parameter $b = 1/\sqrt{2}$. Again, using the finite-size scaling form (see text and eq. (29)) with $\beta = 1/2$. The theoretical scaling function $F_{20}^{(1)}(z)$ is obtained from the numerical Laplace inversion of eq. (3), setting $K = 20$ and $\alpha = 1$.

To be more concrete, we use a Markov chain $\boldsymbol{\eta}(t) = \boldsymbol{\eta}(0), \boldsymbol{\eta}(1), \ldots$. Every move $\boldsymbol{\eta}(t) \to \boldsymbol{\eta}(t+1)$ consists of changing one entry of $\boldsymbol{\eta}(t)$ leading to a trial $\boldsymbol{\eta}'$ ("local update"). While the simplest method to change would be the replacement of one uniform-distributed random number by a freshly drawn one, as used in ref. [12], this will lead to difficulties especially for small values K. For the far tails, there will be a point where all entries of $\boldsymbol{\eta}$ are almost one (or almost zero) and almost every new proposal will be rejected, since it is improbable to draw a random number very close to the previous one. Therefore, we perform a slightly more involved protocol, where instead of redrawing we change an entry $\eta_i \to \eta_i + \xi\delta$, where $\xi \in [-1, 1]$ is uniformly distributed and $\delta \in \{10^{-i} | i \in \{0, 1, 2, 3, 4, 5\}\}$ with uniform probability 1/6. Thus, δ determines the scale of the local change. Changes resulting in an entry $\eta_i \notin [0, 1)$ are directly *rejected*, *i.e.*, $\boldsymbol{\eta}(t+1) = \boldsymbol{\eta}(t)$. Changes are *accepted*, *i.e.*, $\boldsymbol{\eta}(t+1) = \boldsymbol{\eta}'$, with the Metropolis acceptance ratio $p_{\mathrm{acc}} = \min\{1, \mathrm{e}^{-\Delta E_0/T}\}$, where ΔE_0 is the change in energy caused by the proposed change, and otherwise also rejected.

Sampling this Markov chain at different temperatures, results in a histogram $P_T(E_0)$ for each temperature, which can be corrected for the bias using

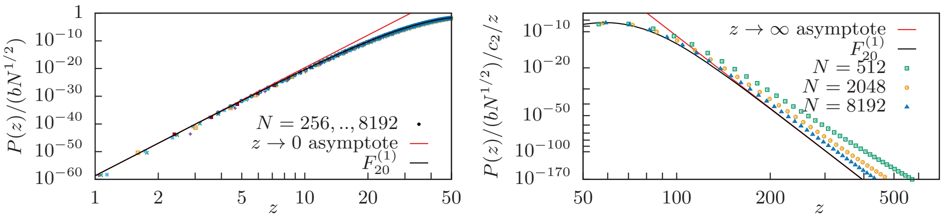
$$P(E_0) = \mathrm{e}^{E_0/T} Z(T) P_T(E_0). \tag{28}$$

The *a priori* unknown normalization parameter $Z(T)$ can be obtained by enforcing continuity and normalization of the whole distribution, which is obtained from performing simulations for several values of T, including $T = \infty$, which corresponds to simple sampling. We will not go into further details, since this algorithm is well described in several other publications [11–13].

For the Pareto distributed case $p(\varepsilon) = \frac{2}{\varepsilon^3}\Theta(\varepsilon - 1)$, we used instead a modified *Wang-Landau* sampling [14,15] with subsequent entropic sampling [16,17]. We used Wang-Landau sampling for this case, since the temperatures are harder to adjust, *i.e.*, for negative temperatures it happens quickly that equilibration becomes impossible and the energy increases constantly. This effect is already known to pose difficulties for the aforementioned sampling with bias [18,19].

We set $K = 20$ in fig. 1 and compare the distribution $P_{K=20,N}(E_0)$ for different values of N. We verify, by a data collapse, the scaling form predicted in eq. (2) and also compare the numerical scaling function to the analytical ones.

While the exponential case fits very well to the analytic result even for small values of N, the other cases show strong finite-size effects especially in the extreme right tail. Such finite-size effects are known to occur frequently in the extreme statistics of i.i.d. random variables [20]. As seen in fig. 1, the discrepancy between the numerical and the analytical results is very small in the main region (*i.e.*, in the bulk). In the tails, we need to use a finite-size ansatz to study the convergence of the numerical results as $N \to \infty$. For example, it is natural to expect that the finite-size corrections to the leading scaling form in eq. (2)



(a) left tail

(b) right tail

Fig. 2: (Colour online) We consider the tails of the distribution $P_{K=20,N}(E_0)$ for the case $p(\varepsilon) = \varepsilon e^{-\varepsilon}$, corresponding to $\alpha = 1$. The scaling function in this case is given by $F^{(1)}_{K=20}(z)$. The asymptotic behaviours of $F^{(1)}_{K=20}(z)$ in eq. (4), for $z \to 0$ (left tail, in (a)) and $z \to \infty$ (right tail in (b)) are compared to numerical simulations. The data have been plotted on a scale such that the two cases from eq. (4) appear as straight lines. In (b), for clarity, only three different values of N have been plotted.

are of the form

$$P_{K,N}(E_0) = b\, N^{1/(1+\alpha)} \times \left[F^{(\alpha)}_K(z) + N^{-\beta} G^{(\alpha)}_K(z) + N^{-2\beta} H^{(\alpha)}_K(z) + \dots \right], \quad (29)$$

where $\beta = \min(1/(1+\alpha), 1)$, $z = b\, N^{1/(\alpha+1)} E_0$ is the scaling variable, and $G^{(\alpha)}_K(z), H^{(\alpha)}_K(z)$ describe the finite-size scaling of the correction terms. Thus, for $\alpha = 0$ one has $\beta = 1$, while for $\alpha = 1$, we have $\beta = 1/2$. For several values of z, we extrapolate the data by fitting pointwise the numerical data in fig. 1 as a function of N, to obtain estimates for the asymptotes $F^{(\alpha)}_K(z)$. We treated the cases c) (corresponding to $\alpha = 0$, and hence $\beta = 1$) and d) ($\alpha = 1$, $\beta = 1/2$), the extrapolated values are shown as symbols. Furthermore, in fig. 2, we show the behaviour in the tails for the case d), which exhibits the strongest finize-size effects, such that the asymptotic behaviour eq. (4) is directly visible. It is apparent that the convergence for large values of N is faster in the left tail $z \to 0$, while it is much slower in the right tail $z \to \infty$.

Conclusion. – In this paper, we have studied analytically and numerically the full distribution of the ground-state energy of K non-interacting fermions in a disordered environment, modelled by a Hamiltonian whose spectrum consists of N i.i.d. random energy levels with distribution $p(\varepsilon)$ (with $\varepsilon \geq 0$), in the same spirit as the "Random Energy Model". This ground-state energy is the sum of the smallest K values drawn from a probability distribution and, therefore, a generalization of the extreme-value statistics, which corresponds to the case $K = 1$. Thus, our results should be of interest also in a very general mathematical context.

We have shown that for each fixed K, the distribution $P_{K,N}(E_0)$ of the ground-state energy has a universal scaling form in the limit of large N (see eq. (2)). This universal distribution depends only on K and the exponent α characterizing the small-ε behaviour of $p(\varepsilon) \sim \varepsilon^\alpha$. We derive an exact expression for the Laplace transform of this scaling function in eq. (3). For generic α, the asymptotic behaviors of the scaling function are derived explicitly in eq. (4), while for the special case $\alpha = 0$, the Laplace transform can be explicitly inverted, giving the full scaling function in eq. (24). Numerically, while the peak region of the distribution of E_0 can be easily estimated by standard methods, estimating the tails of the distribution where the probability is very small is hard and requires more sophisticated techniques. In this paper, using an importance sampling algorithm, we were able to estimate the tail probabilities (up to a precision as small as 10^{-160}) and thereby to verify the theoretical predictions. Thus, the main conclusion of our work is that, even though the individual energy levels are independent random variables, the ordering needed to compute the ground-state energy induces effective correlations between the energy levels. These effective correlations then lead, for the ground-state energy, to a whole new class of universal scaling functions parameterised by K and α.

In this work, we have modelled the single-particle energy levels of a quantum disordered system by i.i.d. random variables, *à la* REM. This REM approximation for the energy levels is known to be valid for disordered Hamiltonians whose eigenstates are strongly localised in space [2]. Thus, we expect that the results presented in this paper for the universal distribution of the ground-state energy would apply to such strongly disordered quantum systems. It is then natural to ask what happens to the ground-state energy for Hamiltonians with weakly localised eigenstates. In some weakly localised systems, a description based on Random Matrix Theory (RMT) [2] is a good approximation, where the energy levels (identified with the eigenvalues of a random matrix) are strongly correlated with mutual level repulsion. In this RMT context, several linear statistics of ordered eigenvalues have been recently introduced and studied for large N under the name of truncated linear statistics (TLS) [21,22]. The ground-state energy in eq. (1) or more generally the linear statistics as in eq. (12) studied here are instances of TLS, but for i.i.d. random variables. It would thus be interesting to see how the TLS, studied here for i.i.d. variables, crosses over to the RMT case, as one goes from the strongly localised part of the spectrum of a disordered Hamiltonian to the weakly localised part.

* * *

We acknowledge the HPC facilities of the GWDG Göttingen and the CARL cluster in Oldenburg funded by the DFG (INST 184/157-1 FUGG) and the Ministry of Science and Culture (MWK) of the Lower Saxony State. HS and AKH thank the LPTMS for hospitality and financial support. HS acknowledges support by DFG grant HA 3169/8-1. SNM and GS acknowledge support by ANR grant ANR-17-CE30-0027-01 RaMaTraF.

REFERENCES

[1] Derrida B., *Phys. Rev. B*, **24** (1981) 2613.

[2] Moshe M., Neuberger H. and Shapiro B., *Phys. Rev. Lett.*, **73** (1994) 1497.

[3] Gumbel E. J., *Statistics of Extremes* (Dover, New York) 1958.

[4] David H. A. and Nagaraja H. N., *Order Statistics*, third edition (John Wiley & Sons, New York) 2003.

[5] Nagaraja H. N., *Ann. Inst. Stat. Math.*, **33** (1981) 437.

[6] Nagaraja H. N., *Ann. Stat.*, **10** (1982) 1306.

[7] Johansson F. *et al.*, mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 1.0.0) (2013), `http://mpmath.org/`.

[8] Press W. H., Teukolsky S. A., Vetterling W. T. and Flannery B. P., *Numerical Recipes: The Art of Scientific Computing*, 3rd edition (Cambridge University Press) 2007.

[9] Corless R. M., Gonnet G. H., Hare D. E. G., Jeffrey D. J. and Knuth D. E., *Adv. Comput. Math.*, **5** (1996) 329.

[10] Gough B., *GNU Scientific Library Reference Manual*, 3rd edition (Network Theory Ltd) 2009.

[11] Hartmann A. K., *Phys. Rev. E*, **65** (2002) 056102.

[12] Hartmann A. K., *Phys. Rev. E*, **89** (2014) 052103.

[13] Schawe H., Hartmann A. K. and Majumdar S. N., *Phys. Rev. E*, **97** (2018) 062159.

[14] Wang F. and Landau D. P., *Phys. Rev. Lett.*, **86** (2001) 2050.

[15] Belardinelli R. E. and Pereyra V. D., *Phys. Rev. E*, **75** (2007) 046701.

[16] Lee J., *Phys. Rev. Lett.*, **71** (1993) 211.

[17] Dickman R. and Cunha-Netto A. G., *Phys. Rev. E*, **84** (2011) 026701.

[18] Claussen G., Hartmann A. K. and Majumdar S. N., *Phys. Rev. E*, **91** (2015) 052104.

[19] Schawe H., Hartmann A. K. and Majumdar S. N., *Phys. Rev. E*, **96** (2017) 062101.

[20] Gyorgyi G., Moloney N. R., Ozogany K., Racz Z. and Droz M., *Phys. Rev. E*, **81** (2010) 041135.

[21] Grabsch A., Majumdar S. N. and Texier C., *J. Stat. Phys.*, **167** (2017) 234.

[22] Grabsch A., Majumdar S. N. and Texier C., *J. Stat. Phys.*, **167** (2017) 1452.

A.5. Large deviations of the length of the longest increasing subsequence of random permutations and random walks

The first author Jörn Börjes worked as part of his Bachelor's thesis at the University of Oldenburg in the working group of Alexander K. Hartmann on the distributions of the longest increasing subsequences in variously constructed sequences. Hendrik Schawe, is the author of the thesis at hand. Alexander K. Hartmann is the supervising professor of H. Schawe.

This project was conceived at the LPTMS after a talk of J. Ricardo G. Mendonça about a similar topic in a discussion between S. N. Majumdar, A. K. Hartmann and H. Schawe. After a preliminary feasibility test of H. Schawe, J. Börjes started working on the problem advised by H. Schawe and A. K. Hartmann. The first draft was prepared by J. Börjes and improved in some iterations by all authors.

PHYSICAL REVIEW E **99**, 042104 (2019)

Large deviations of the length of the longest increasing subsequence of random permutations and random walks

Jörn Börjes,* Hendrik Schawe,† and Alexander K. Hartmann‡
Institut für Physik, Universität Oldenburg, 26111 Oldenburg, Germany

(Received 17 January 2019; published 2 April 2019)

We study numerically the length distribution of the longest increasing subsequence (LIS) for random permutations and one-dimensional random walks. Using sophisticated large-deviation algorithms, we are able to obtain very large parts of the distribution, especially also covering probabilities smaller than 10^{-1000}. This enables us to verify for the length of the LIS of random permutations the analytically known asymptotics of the rate function and even the whole Tracy-Widom distribution. We observe a rather fast convergence in the larger than typical part to this limiting distribution. For the length L of LIS of random walks no analytical results are known to us. We test a proposed scaling law and observe convergence of the tails into a collapse for increasing system size. Further, we obtain estimates for the leading-order behavior of the rate functions in both tails.

DOI: 10.1103/PhysRevE.99.042104

I. INTRODUCTION

We study the length distribution of the *longest increasing subsequence* (LIS) [1] of different ensembles of random sequences. A subsequence of a sequence S consists of elements of S in the same order as in S. But neighbors in the subsequence are not necessarily neighbors in S. For a LIS it is required that the elements of the subsequence are increasing from left to right, and the number of elements in the subsequence is maximal.

The first mention of this problem involving random permutations (RPs) is from Stanisław Ulam [2] and is also known as "Ulam's problem." In his study the mean length L of LIS on RP of n integers were examined by means of Monte Carlo simulations. It was conjectured that, in the limit of large n, the length converges to $L = c\sqrt{n}$, with a constant c, which was later proven to be $c = 2$ [3]. In the following years much work was published scrutinizing the large-deviation behavior of this problem, and explicit expressions for both the left (lower) and right (upper) tail were derived rigorously [4–6]. Interestingly, for the LIS of RPs it was shown that the length distribution $P(L)$ is a Tracy-Widom distribution [7].

The Tracy-Widom distribution was at that time only known from random matrix theory, where it describes the fluctuations of the largest eigenvalues of the *Gaussian unitary ensemble* (GUE), an ensemble of Hermitian random matrices. In physics it came into focus after an explicit mapping of an $1 + 1$-dimensional polynuclear growth model [8]. Subsequently other mappings of $1 + 1$-dimensional growth models belonging to the Kardar-Parisi-Zhang universality like an anisotropic ballistic deposition were found [9]. Other models, in which the Tracy-Widom distribution appears, include the totally asymmetric exclusion process [10] and directed polymers [11]. For a pedagogical overview of the relations of different models exhibiting a Tracy-Widom distribution there are some review articles, e.g., Refs. [12–14]. Fluctuations in growth processes following the Tracy-Widom distribution could also be observed in experiments, e.g., from growing liquid crystals where the Tracy-Widom distribution of the GUE appears for circular growth [15] and of the *Gaussian orthogonal ensemble* (GOE) for growth from a flat surface [16].

The Tracy-Widom distribution seems to occur always together with a *third-order phase transition* between a *strongly interacting* phase in the left tail and a *weakly interacting* phase in the right tail [17]. For these third-order phase transitions, the probability density function behaves in the left tail as $P(x) \approx e^{-n\Phi_-}$ with the role of the free energy played by the *rate function* $\Phi_-(x) \sim (a - x)^3$ for $x \to a$ from the left, where the scaled mean value a is the critical point of the transition. Here n is some large parameter, e.g, the system size. The $O(x^3)$ leading-order behavior of Φ_- generally leads to a discontinuity in the third derivative of the free energy and therefore to a third-order phase transition. This seems to be a characteristic sign predicting the main region of the distribution to follow a Tracy-Widom distribution. Therefore the behavior of the far tails of these problems is of great interest to understand this connection better. Consequently the *large deviations* of some of these models were studied thoroughly [17,18].

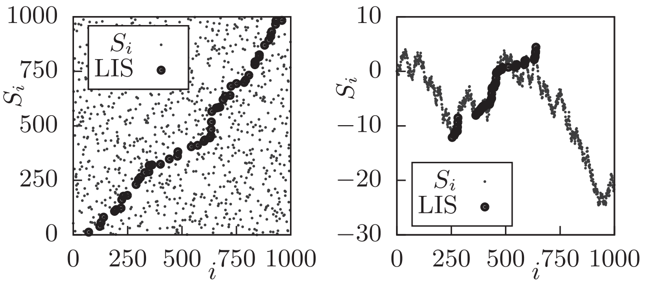
For the length distribution in the RP case the large deviations, the behavior for large values of n including the far tails, are known analytically [4–7]. These show the characteristic behavior of the above mentioned left-tail rate function. For the case of random walks (RWs), bounds for the behavior of the mean are known [19], and there is also numerical work which is concerned with the distribution in the typical region [20], i.e., those LISs which occur with a high enough probability of about $\geqslant 10^{-6}$. We also deem it worthwhile to look closer at the tails of the distribution for finite systems.

For the purpose of studying the large deviations of this problem numerically, we utilize sophisticated large-deviation

---

*joern.boerjes@uni-oldenburg.de
†hendrik.schawe@uni-oldenburg.de
‡a.hartmann@uni-oldenburg.de

©2019 American Physical Society



(a) (b)

FIG. 1. Visualization of random sequences of length $n = 1000$ where the value is plotted over the corresponding index. Marked with circles are the entries of one possible LIS. (a) Random permutation (RP), (b) random walk (RW).

sampling methods to observe the distribution $P(L)$. In this way we can observe directly the far tails of the Tracy-Widom distribution for the RP case [7] and can confirm the known large n asymptotics [6]. The second ensemble are one-dimensional RWs with increments from a uniform distribution. While we can observe the scaling proposed in Ref. [20] for the main region, the tails are subject to considerable finite-size effects. Nevertheless the distributions collapse over larger regions for larger sizes n. Also, we give estimates for the leading-order behavior of the rate functions governing the left and right tails of the distribution $P(L)$.

This study first introduces the different ensembles of interest and the algorithms used to obtain the distribution of the length in Sec. II. Section III shows the results we gathered and interprets them. We conclude this study in Sec. IV.

II. MODELS AND METHODS

To define the LIS, we have to define a subsequence first. Given some sequence $S = (S_1, S_2, \dots, S_n)$ a *subsequence* of length L is a sequence $s = (S_{i_1}, S_{i_2}, \dots, S_{i_L})$ $(1 \leqslant i_j \leqslant n, i_j < i_{j+1}$ for all $j = 1, \dots, L)$ containing only elements present in S in the same order as in S, though possibly with gaps. An *increasing subsequence* has elements such that every element in s is smaller than its predecessor, i.e., $S_{i_j} < S_{i_{j+1}}$ for $j = 1, \dots, L - 1$. The LIS is consequently the longest, i.e., the one with the highest number L of elements, of all possible increasing subsequences. Note that the LIS is not necessarily unique, but by definition its length is unique. As an example two different LISs are marked by overlines and underlines in the following sequence: $S = (\underline{3}, 9, \underline{4}, \overline{1}, \overline{2}, \underline{7}, \overline{6}, \underline{\overline{8}}, 0, 5)$.

In this study the sequence S is drawn either from the ensemble of *random permutations* of n consecutive integers or from the ensemble of *random walks* with increments δ_j $(j = 1, \dots, n)$ from a uniform distribution $\delta_j \sim U(-1, 1)$, such that

$$S_i = \sum_{j=1}^{i} \delta_j. \tag{1}$$

An example of each sequence with the corresponding LIS marked is shown in Fig. 1.

To find L of any given sequence, we use the *patience sort algorithm*. We introduce only the very simple version to obtain the length, but a comprehensive review of the connection of patience sort with the LIS can be found in Ref. [3]. In short, the patience sort algorithm works as follows: We iterate over the n entries S_i and place each into an initially empty stack (or pile) a_j on the smallest j such that for the top entry $\mathrm{top}(a_j) > S_i$ holds. Note that this always ensures that the top entries of a are ascendingly sorted, such that we can determine j by a binary search in $O(\ln n)$. Finally, the number of nonempty stacks a_j is equal to the length L of the LIS.

A. Large-deviation sampling

To be able to gather statistics of the large-deviation regime numerically [21], we need to apply a sophisticated sampling scheme. Therefore we use a well-tested [22–24] Markov chain Monte Carlo sampling which treats the system as a canonical system at an artificial *temperature* with the observable of interest as its *energy*. Since the algorithm has been presented comprehensively in the literature, we here mainly state the details specific to the current application. In our case, we identify the state of the system with the sequence, the length L with the energy and sample the equilibrium state at temperature Θ using the Metropolis algorithm [25,26]. Controlling the temperature allows us to direct the sampling to different regimes of the distributions, to eventually cover the distribution over a large part of the support. To evolve our Markov chain of sequences, we have to introduce change moves, which modify a sequence and consequently the energy L. For the RP we swap two random entries, and for the RW we replace one of the increments δ_j [cf. Eq. (1)] by a new random number drawn from the same uniform distribution. These changes are accepted according to the Metropolis acceptance ratio

$$P_{\mathrm{acc}} = \min(1, e^{-\Delta L/\Theta}), \tag{2}$$

where ΔL is the change in energy due to the change move. This Markov chain of sequence realizations converges to an equilibrium state. As usual with Markov chain Monte Carlo simulations, we need to ensure equilibration and that the samples are decorrelated [26].

In equilibrium the realizations generally have a lower than typical energy for low temperatures and typical energies for high temperatures. We also introduce negative temperatures for larger than typical energies. This way the temperature can be tuned to guide the simulation towards realizations within a specific range of energies L. We know the equilibrium distribution $Q_\Theta(S)$ at temperature Θ of realizations, i.e., sequences S, to be

$$Q_\Theta(S) = \frac{1}{Z_\Theta} e^{-L(S)/\Theta} Q(S), \tag{3}$$

with the natural distribution $Q(S)$, i.e., the distribution of realizations arising by simply generating subsequences uniformly. This can be exploited to correct for the bias introduced by the temperature and arrive at the unbiased distribution $P(L)$ with good statistics also in the regions unreachable by simple sampling. Therefore consider the sampled equilibrium distributions $P_\Theta(L)$. To connect them to the distribution of realizations $Q_\Theta(S)$, we can sum all realizations with the same

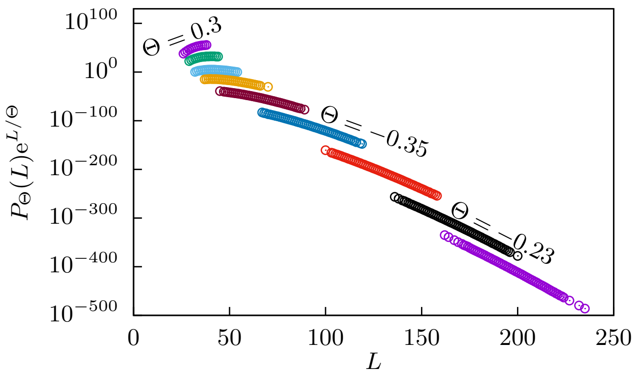


FIG. 2. Intermediate step after correction with Eq. (6) but before determination of the values Z_{Θ_i} (i.e., all $Z_{\Theta_i} = 1$). The data are gathered for RW sequences of length $n = 512$. Each shade of gray (color) is sampled at a different temperature Θ, and for three data sets the corresponding temperatures are annotated. (For clarity some evaluated temperatures are omitted.)

value of L, leading to

$$P_\Theta(L) = \sum_{\{S|L(S)=L\}} Q_\Theta(S) \tag{4}$$

$$= \sum_{\{S|L(S)=L\}} \frac{1}{Z_\Theta} e^{-L(S)/\Theta} Q(S) \tag{5}$$

$$= \frac{1}{Z_\Theta} e^{-L(S)/\Theta} P(L). \tag{6}$$

Solving this equation for $P(L)$ allows us to correct for the bias introduced by the temperature. An intermediate snapshot of this process is shown in Fig. 2.

The constants Z_Θ can be obtained by enforcing continuity of the distribution,

$$P_{\Theta_j}(L)e^{L/\Theta_j} Z_{\Theta_j} = P_{\Theta_i}(L)e^{L/\Theta_i} Z_{\Theta_i}, \tag{7}$$

for pairs of i, j for which the gathered data $P_{\Theta_i}(L)$ overlap with $P_{\Theta_j}(L)$. While this can be used to approximate the ratios of pairwise Z_{Θ_i}, the absolute value can then be obtained by normalization of the whole distribution. This procedure requires a clever choice of temperatures, since gaps in the sampled range of L would make it impossible to find a ratio of Z_{Θ_i} on the left and right sides of the gap. We use on the order of 100 distinct temperatures. In general, the larger the size n, the more temperatures are needed.

III. RESULTS

Before we look into the large-deviation tails, we in brief present some simple sampling results addressing the qualitative difference of RP and RW cases, which are visible in Fig. 1. The entries of the RW are strongly correlated such that the RW typically consists of runs with downward or upward trends. This means that the LIS is typically confined in an upward trend, and its entries therefore are close together. The RP, on the other hand, typically shows LISs with entries over the whole range.

To quantify this effect we measure the fraction of the sequence over which the LIS spans. For multiple system sizes $1024 \leqslant n \leqslant 524\,288$, 10^6 samples each, we measure the positions i, j of the first and last entries of a found LIS to calculate its relative *span* $\rho = (j - i)/n$. We extrapolate the mean span with an offset power law $\rho = an^b + \rho_\infty$ to extrapolate the asymptotic span ρ_∞, which is shown in Fig. 3.

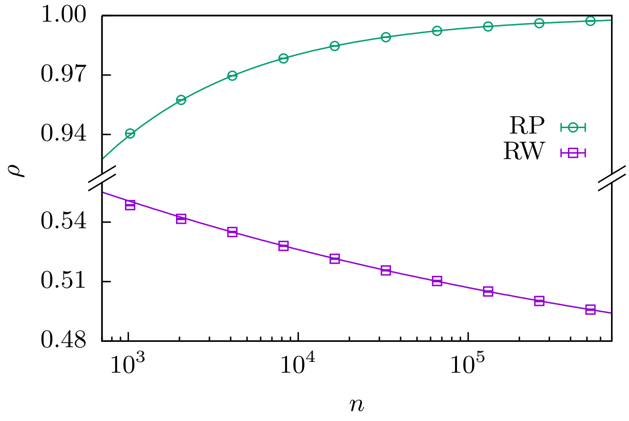


FIG. 3. Extrapolation of the span ρ. Measurements at different sizes n are used to extrapolate an asymptotic span according to a power law with offset $\rho = an^b + \rho_\infty$. Fits to this expression for $n \geqslant 4096$ are marked by a line. The two obtained asymptotic values are $\rho_\infty^{\mathrm{rp}} = 1.00005(2)$ for the RP and $\rho_\infty^{\mathrm{rp}} = 0.439(7)$ for the RW. Note the broken ρ axis. Error bars are smaller than the width of the line.

For the RP we get a value of $\rho_\infty^{\mathrm{rp}} = 1.00005(2)$ and for the RW $\rho_\infty^{\mathrm{rp}} = 0.439(7)$. Note that these numbers are subject to two sources of systematical errors, which can explain, e.g., the impossible result of $\rho_\infty^{\mathrm{rp}} > 1$. First, the function we use to extrapolate is an ansatz, which considers only leading-order behavior of the actual scaling function. Second, we obtain only one LIS per sequence via the backpointer extension of patience sort [3], which might result in a biased selection of LISs. Both questions merit further research on their own but are beyond the scope of this article. This means that LISs of RPs typically span the whole sequence, while LISs of RWs typically span only less than half of the sequence, such that its entries are closer together.

To gather statistics of L, we apply the temperature-based sampling scheme for the two cases of RPs and of RWs with uniform increments. In both cases, we study five different system sizes n up to $n = 4096$ each.

A. Random permutations

First, we look at the LIS length distribution of RPs. For this case there are already many properties known in the limit of $n \to \infty$.

It is known that the distribution should converge to a suitably rescaled Tracy-Widom distribution χ of the GUE ensemble [7] for large values of n as

$$P_n[(L - 2\sqrt{n})n^{-1/6}] = \chi[(L - 2\sqrt{n})n^{-1/6}]. \tag{8}$$

Rescaled to compensate for this leading behavior, our results are shown in Fig. 4. By using the large-deviation approach, we are able to measure probabilities as small as 10^{-1000} and less, allowing us to go beyond the first numerical work [20] on the distribution of LISs. We can observe a very good collapse up

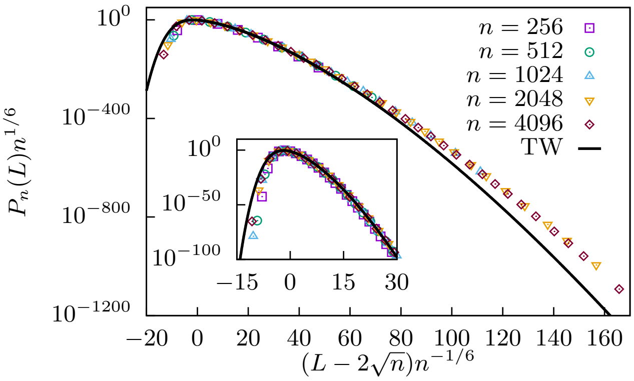


FIG. 4. Numerically obtained distributions for different system sizes n rescaled according to Eq. (8). The Tracy-Widom distribution is drawn as a black line [27] and is expected to be the curve all distributions collapse onto. The inset shows a zoom on the intermediate tails. On the left the tendency of our data towards the Tracy-Widom distribution with increasing system size n is visible. (For clarity some data points are discarded to show the same density of symbols for every system size.)

to probabilities of 10^{-200} of our data onto the Tracy-Widom distribution given in the tables of Ref. [27].

Also note that the collapse works very well in the intermediate right tail but converges a bit slower in the left tail and far slower in the far-right tail. The inset zooms into the intermediate tail of the probability density function $P > 10^{-100}$, where the collapse fits very well to the expected Tracy-Widom distribution. In the far tails we observe considerable deviations, from the tabulated data, which are at least in part caused by finite-size effects due to the relatively small sizes n of our sequences. For a more extensive study of these finite-size effects, one could obtain the empirical distribution for more sizes, and extrapolate the finite-size effects to $n \to \infty$, as done in Ref. [28]. We do not attempt this analysis here, since the very small deviations between different values of n in the right tail suggest that data for much larger sizes would be needed for a meaningful extrapolation. This is at the moment not computationally feasible for us. Nevertheless, our numerically obtained tails fit very well to another expected form, which will be explained later, such that we assume a stronger influence of corrections to scaling in the far tails instead of systematic errors in our data.

Also note that while we can sample a very large part of the distribution $P(L)$ in the RP case—even including events with a probability less than 10^{-1000} for the largest systems—we cannot reach across the whole range of possible values. Possible approaches to extend this range are improvements to our sampling algorithm by, e.g., switching to a better change move or trying a different sampling algorithm like Wang-Landau's method [29].

The left-tail asymptotic, i.e., $L/\sqrt{n} = x < 2$, of the probability density function is given by the analytically known rate function [5,6]

$$\lim_{n\to\infty} \frac{1}{n} \ln P_n(L) = -2H_0(x) \tag{9}$$

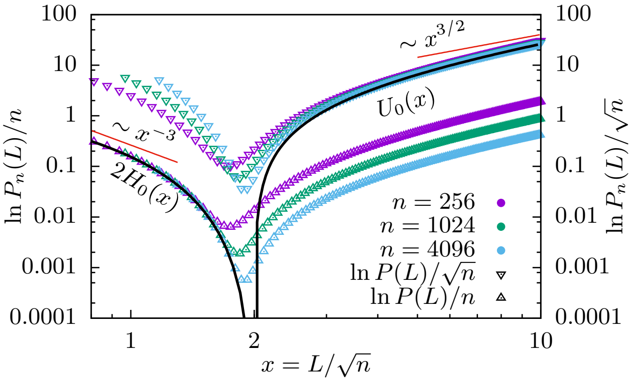


FIG. 5. Empirical rate functions for different system sizes n. On the top (triangles down) scaled as $\ln P_n(L)/\sqrt{n}$ to emphasize the right-tail behavior. On the bottom (triangles up) scaled as $\ln P_n(L)/n$ to emphasize the left tail behavior. The analytically known rate functions for both tails $2H_0$ and U_0 are shown in the correspondingly scaled region and a convergence of the data to these functions is well visible. The leading-order terms of the series expansion (cf. Ref. [4]) are also shown as straight lines next to the rate function.

with

$$H_0(x) = -\frac{1}{2} + \frac{x^2}{8} + \ln\frac{x}{2} - \left(1 + \frac{x^2}{4}\right)\ln\left(\frac{2x^2}{4+x^2}\right); \tag{10}$$

the right-tail asymptotic, i.e., $L/\sqrt{n} = x > 2$, is given by [4,6]

$$\lim_{n\to\infty} \frac{1}{\sqrt{n}} \ln P_n(L) = -U_0(x) \tag{11}$$

with

$$U_0(x) = 2x\cosh^{-1}(x/2) - 2\sqrt{x^2 - 4}. \tag{12}$$

Note that Eq. (11) behaves atypically for a rate function as the distribution behaves like $P_n \propto e^{-\sqrt{n}U_0}$, which according to the definition (e.g., Ref. [30]) does therefore not fulfill the large-deviation principle. Nevertheless, it describes the right-tail behavior of the distribution in leading order.

We use our sampled data to test these rate functions. If the data are suitably rescaled according to Eqs. (9) and (11), in the corresponding tails we can observe a very nice convergence of the data to the rate functions. This is plotted in Fig. 5. This excellent agreement of analytical and numerical results over hundreds of decades in probability gives us confidence that our approach works well and can be extended to cases where no analytical results are known. Also note that we can observe in our data the leading-order behavior of the left-tail rate function H_0, which goes with the exponent 3 characteristic for the third-order phase transition, confirming its connection with the Tracy-Widom distribution [17].

B. Random walks

The second class of sequences S we scrutinize are RWs. The distribution beyond the high-probability peak region seems to be unknown. Again, by applying the large-deviation approach, we sample basically the whole distribution and can even compare the right tail of our distribution with the

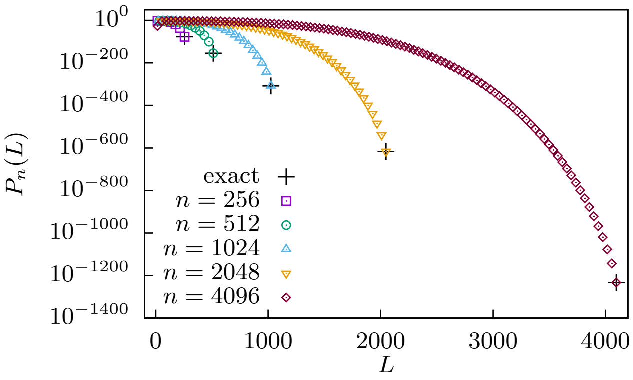


FIG. 6. Probability distributions $P_n(L)$ of the length of the LIS of RWs with exact extreme values for the $n = L$ case. (For clarity only every 40th bin is visualized, including the $n = L$ bin.)

corner case of $L = n$, which occurs only if all increments δ are positive and therefore with probability 2^{-n}. This case is marked in Fig. 6 to emphasize the quality of our data. For the left tail, we can not sample so far, as the very steep decline of the distribution is difficult to handle for our sampling scheme.

For RWs with increments from a symmetric uniform distribution, indeed for increments from any symmetric distribution with finite variance, the scaling of the mean as $\langle L\rangle \propto n^\theta$ and the variance as $\sigma^2 \propto n^{2\theta}$ was observed in Ref. [20] with $\theta = 0.5680(15)$ for finite system sizes. This observation lead to the assumption that the whole distribution follows the scaling form

$$P_n(L) = \langle L\rangle g(\langle L\rangle L), \tag{13}$$

with a not explicitly known function g. Even more, Ref. [20] suggests that their measurements can be explained, instead of the exponent θ, by a logarithmic correction to a square-root scaling:

$$\langle L\rangle \approx \frac{1}{e}\sqrt{n}\ln n + \frac{1}{2}\sqrt{n}. \tag{14}$$

Using our data for the tails of the distribution, we can test whether this scaling holds over the whole distribution or only in the main region. If we rescale the axis of the plot suitably, the distributions for different sizes n should collapse on the scaling function g, in the case that Eq. (13) holds. Note that the related problem of the longest weakly increasing subsequence for RW increments of ±1 is known to scale also with $\sqrt{n}$ but does not exhibit the logarithmic correction [19]. Our collapse in Fig. 7 following Eq. (13) supports the validity of Eq. (14). The collapse does work except for the very far tails, which is an effect—at least partially—caused by finite-size effects, since the length of the LIS can for finite n never be longer than n. This pattern occurs often when looking at the far tails of discrete systems, e.g., for the convex hull of RWs on lattices in Refs. [31–34] or in a toy model for noninteracting Fermions in a landscape with n random energy levels [28].

Since for the rate functions characterizing the LIS length distribution of RWs there is no known result, we use our numerical data to give a rough estimate of the rate function. Therefore we look into the empirical rate function $\Phi_n(L) = \frac{1}{n}\ln P_n(L)$, which is plotted in Fig. 8 for the data already shown in Fig. 6.

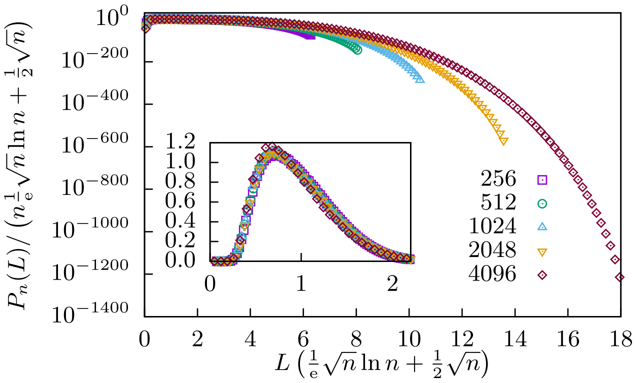


FIG. 7. Collapse of different system sizes on a common curve g from Eq. (13), with $\langle L\rangle$ given by Eq. (14). Apparently the far tail shows corrections to the proposed scaling for finite sizes, which are explained by finite-size effects, e.g., that there is a maximum length of n for finite systems. For increasing sizes n a convergence to a common curve is visible. The inset shows the same in linear scale around the maximum. (For clarity not all data points are drawn.)

Using the empirical rate function we can obtain the asymptotics of the rate function from our data. Note that to estimate the right-tail rate function we use the intermediate tail and not the far tail, which is bending up due to finite-size effects, as the very long LISs are suppressed by the hard limit of $L \leqslant n$.. Since we are interested only in the leading-order exponent of the rate function, i.e., assuming $\Phi(L) \propto L^\kappa$ for very small and very large values of L, we can rescale the axes arbitrarily due to the scale invariance of power laws. For convenience we look at $x = L/L_{\max}$ to limit the range to the interval [0,1]. For the left tail we observe a leading-order behavior of the rate function of approximately $\Phi(L) \sim L^{-1.6}$ and for the right tail $\Phi(L) \sim L^{2.9}$. Note that the exponent of the left tail is clearly distinct from 3, such that it does not show signs of a third-order phase transition. Also it does not show a Tracy-Widom distribution in the main region (also see Ref. [20]), which is

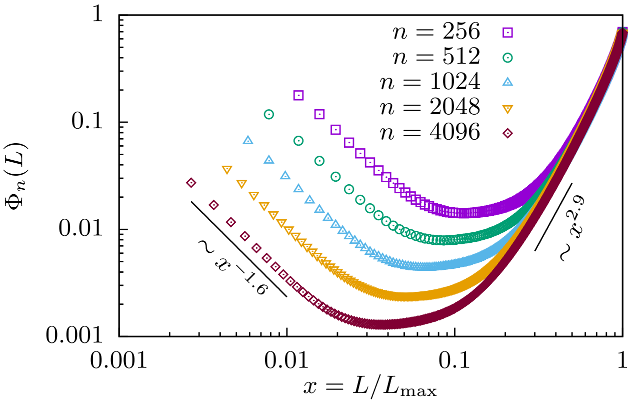


FIG. 8. Empirical rate function $\Phi_n(L)$ for the length of the LIS of RWs. The two straight lines are obtained by power-law fits and show the leading-order behavior of the rate function for each tail. (For clarity not every data point is shown.)

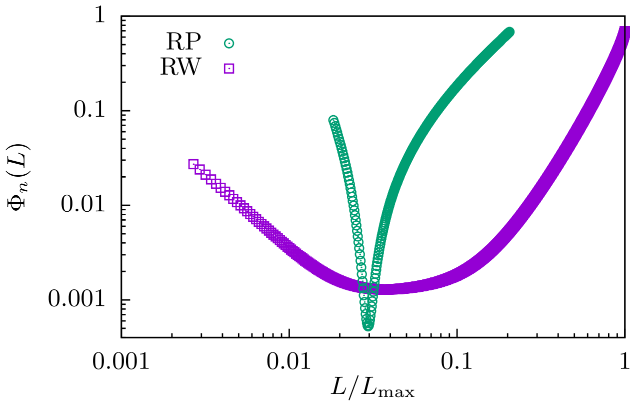


FIG. 9. Direct comparison of the empirical rate function $\Phi_{4096}(L)$ of the RP and the RW.

consistent with the expectation that these two properties do occur together [17].

A comparison of this leading-order behavior to the behavior of the RP case, as visualized in Fig. 5, shows that the tails decay differently. For a direct comparison of our results consider Fig. 9. While the right-tail exponent is larger in the RW case, the probability density decays slower (i.e., the empirical rate function increases slower). This apparent contradiction is understandable when considering that the rate function of the RP case grows much faster near the minimum at $\langle L\rangle$ before it settles into the asymptotic behavior. The RW case behaves exactly opposite, such that the branches left and right of the minimum show opposite curvature in the two cases. Generally, this leads to a distribution $P(L)$ which is much broader in the RW case, especially towards quite large values of L.

IV. CONCLUSIONS

We obtain numerical data for the distribution of the length of the longest increasing subsequence for two cases of sequences of random numbers, namely, for RPs and for one-dimensional RWs. By applying sophisticated large-deviation algorithms, we are able to sample the distributions over literally hundreds of decades in probability. The case of RPs is already well studied analytically in the literature, and we are able to confirm, to the best of our knowledge, for the first time these analytical results. Since our data are gathered for finite system sizes, we can observe a rather fast convergence to the analytical results valid in the $n \to \infty$ limit. These results also show the validity and convergence of our simulations. For the case of RWs we can observe the leading-order behavior of the rate function far into the tails. Also our data support the scaling assumption Eq. (13) [20] for the whole distribution including the logarithmic term, which is not present for weakly increasing subsequences in RWs with ±1 steps [19]. This result could be used to guide analytical work on this topic and to test future analytical results. A direct comparison of the empirical rate functions in the tails shows qualitatively very different behavior. While the rate function of the RW seems to be a convex function, the RP case consists, in principle, of two concave parts.

A possible future direction extending this work would be an interpolation between the RP and RW case, where one could observe the change of the exponents governing the rate function. Since a set of distinct random numbers δ_j drawn uniformly from $[-1, 1]$ should show the same statistics for the longest increasing subsequence of a RP, we could introduce a parameter c governing the correlation length. The sequence would be constructed as $S_i = \sum_{j=\max(0,i-c)}^{i} \delta_j$. For $c = 0$ this would correspond to a RP and for $c = n$ to a RW. In addition to this simple type of correlation, one could study power-law correlated random numbers or increments, leading possibly to even more complicated behavior.

Furthermore, it is of interest to analyze the actual LIS in particular with taking the degeneracy into account. For this purpose one must use a dynamic programming approach, which exhibits a running time of $O(n^2)$ instead of the $O(n \ln n)$ complexity of the algorithm which obtains just the length of the LIS. Nevertheless, the dynamic programming approach would allow one to compare different LISs for every realization of the sequence, whether they are rather similar or possibly very different, depending on the type of sequence. Also one could study the distribution of the LIS entropy with similar large-deviation techniques as applied here. Furthermore, this would allow to measure a correlation between LIS length and span in a statistical unbiased way, going beyond the results shown in Fig. 3.

ACKNOWLEDGMENTS

We are indebted to Satya N. Majumdar, who brought this problem to our attention, and for his feedback on a draft of this manuscript. We also want to thank J. Ricardo G. Mendonça for useful discussions and his feedback on a draft of the manuscript. Further we thank Christoph Norrenbrock for helpful comments and advice during the preparation of the manuscript. We acknowledge usage of the HPC facilities of the GWDG Göttingen and the CARL cluster in Oldenburg funded by the DFG (INST 184/157-1 FUGG) and the Ministry of Science and Culture (MWK) of the Lower Saxony State. H.S. acknowledges support by DFG grant HA 3169/8-1.

---

[1] D. Romik, *The Surprising Mathematics of Longest Increasing Subsequences* (Cambridge University Press, New York, 2015).
[2] S. M. Ulam, in *Modern Mathematics for the Engineer: Second Series*, edited by E. Beckenbach and M. Hestenes, Dover Books on Engineering Series (Dover Publications, New York, 2013), Chap. 11, pp. 261–281.
[3] D. Aldous and P. Diaconis, Bull. Am. Math. Soc. **36**, 413 (1999).
[4] T. Seppäläinen, Probab. Theory Relat. Fields **112**, 221 (1998).
[5] B. F. Logan and L. A. Shepp, Adv. Math. **26**, 206 (1977).
[6] J.-D. Deuschel and O. Zeitouni, Comb. Probab. Comput. **8**, 247 (1999).
[7] J. Baik, P. Deift, and K. Johansson, J. Am. Math. Soc. **12**, 1119 (1999).
[8] M. Prähofer and H. Spohn, Phys. Rev. Lett. **84**, 4882 (2000).

[9] S. N. Majumdar and S. Nechaev, Phys. Rev. E **69**, 011103 (2004).

[10] K. Johansson, Commun. Math. Phys. **209**, 437 (2000).

[11] J. Baik and E. M. Rains, J. Stat. Phys. **100**, 523 (2000).

[12] T. Kriecherbauer and J. Krug, J. Phys. A: Math. Theor. **43**, 403001 (2010).

[13] S. N. Majumdar, in *Complex Systems: Lecture Notes of the Les Houches Summer School 2006*, edited by J. Bouchaud, M. Mézard, and J. Dalibard (Elsevier Science, Les Houches, 2006), Chap. 4, pp. 179–216.

[14] I. Corvin, Random Matrices: Theory Appl. **01**, 1130001 (2012).

[15] K. A. Takeuchi and M. Sano, Phys. Rev. Lett. **104**, 230601 (2010).

[16] K. A. Takeuchi, M. Sano, T. Sasamoto, and H. Spohn, Sci. Rep. **1**, 34 (2011).

[17] S. N. Majumdar and G. Schehr, J. Stat. Mech.: Theory Exp. (2014) P01012.

[18] P. L. Doussal, S. N. Majumdar, and G. Schehr, EPL (Europhys. Lett.) **113**, 60004 (2016).

[19] O. Angel, R. Balka, and Y. Peres, Math. Proc. Cambridge Philos. Soc. **163**, 173 (2017).

[20] J. R. G. Mendonça, J. Phys. A: Math. Theor. **50**, 08LT02 (2017).

[21] A. K. Hartmann, *Big Practical Guide to Computer Simulations* (World Scientific, Singapore, 2015).

[22] A. K. Hartmann, Phys. Rev. E **65**, 056102 (2002).

[23] A. K. Hartmann, Eur. Phys. J. B **84**, 627 (2011).

[24] A. K. Hartmann, Phys. Rev. E **89**, 052103 (2014).

[25] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, J. Chem. Phys. **21**, 1087 (1953).

[26] M. Newman and G. Barkema, *Monte Carlo Methods in Statistical Physics* (Oxford University Press, New York, 1999), pp. 3–86.

[27] M. Prähofer and H. Spohn, J. Stat. Phys. **115**, 255 (2004).

[28] H. Schawe, A. K. Hartmann, S. N. Majumdar, and G. Schehr, EPL (Europhys. Lett.) **124**, 40005 (2018).

[29] F. Wang and D. P. Landau, Phys. Rev. Lett. **86**, 2050 (2001).

[30] H. Touchette, Phys. Rep. **478**, 1 (2009).

[31] G. Claussen, A. K. Hartmann, and S. N. Majumdar, Phys. Rev. E **91**, 052104 (2015).

[32] H. Schawe, A. K. Hartmann, and S. N. Majumdar, Phys. Rev. E **96**, 062101 (2017).

[33] H. Schawe, A. K. Hartmann, and S. N. Majumdar, Phys. Rev. E **97**, 062159 (2018).

[34] H. Schawe and A. K. Hartmann, J. Phys.: Conf. Ser. (to be published).

A.6. Large-deviation properties of the largest biconnected component for random graphs

The first author, Hendrik Schawe, is the author of the thesis at hand. Alexander K. Hartmann is the supervising professor of H. Schawe.

The research question handled in this article was conceived by A. K. Hartmann. Except for data previously published in References [31, 37] (distribution of the size of the single connected components and 2-core for Erdős-Rényi graph) all data was generated by simulation code written by H. Schawe and evaluated by H. Schawe. During regular meetings of A. K. Hartmann and H. Schawe the results were discussed and the scope of the study was refined under consideration of intermediate results. The first draft was prepared by H. Schawe and after a few iterations of feedback from A. K. Hartmann submitted.

Eur. Phys. J. B (2019) 92: 73
https://doi.org/10.1140/epjb/e2019-90667-y

THE EUROPEAN
PHYSICAL JOURNAL B

Regular Article

Large-deviation properties of the largest biconnected component for random graphs

Hendrik Schawe[a] and Alexander K. Hartmann

Institut für Physik, Universität Oldenburg, 26111 Oldenburg, Germany

Received 12 November 2018 / Received in final form 30 January 2019
Published online 3 April 2019
© EDP Sciences / Società Italiana di Fisica / Springer-Verlag GmbH Germany, part of Springer Nature, 2019

Abstract. We study the size of the largest biconnected components in sparse Erdős–Rényi graphs with finite connectivity and Barabási–Albert graphs with non-integer mean degree. Using a statistical-mechanics inspired Monte Carlo approach we obtain numerically the distributions for different sets of parameters over almost their whole support, especially down to the rare-event tails with probabilities far less than 10^{-100}. This enables us to observe a qualitative difference in the behavior of the size of the largest biconnected component and the largest 2-core in the region of very small components, which is unreachable using simple sampling methods. Also, we observe a convergence to a rate function even for small sizes, which is a hint that the large deviation principle holds for these distributions.

1 Introduction

The robustness of networks [1–5] attracted much interest in recent time, from practical applications for, e.g., power grids [6–8], the internet [9,10], to examinations of genomes [11,12]. As typical in network science, one does not only study the properties of existing networks. To model the properties of real networks, different ensembles of random graphs were devised, e.g., Erdős–Rényi (ER) random graphs [13], small world graphs [14], or scale-free graphs [15]. Also for such ensembles the robustness has been studied by analytical and numerical means [16–19]. One often used approach to determine the robustness of networks are *fragmentation* studies, where single nodes are removed from the network. These nodes are selected according to specific rules ("attack") or randomly ("failure"). The functionality, e.g., whether it is still connected, is tested afterwards. It has been suggested that the large deviations are of interest to network robustness, e.g., for the size of the giant connected component, rare configurations of the realization of the damage to networks may change the typically continuous phase transition to a discontinuous phase transition [20,21]. A property necessary for robustness is thus that the graph stays connected when removing an arbitrary node. This exact concept is characterized by the *biconnected component*, which are the connected components which stay connected after an arbitrary node is removed. The existence of a large biconnected component is thus a simple and fundamental property of a graph robust to fragmentation. Another, though related, often studied form of stability looks at the flow through or the transport capability [10] of a graph. Also here a large biconnected component is a good indicator for stability. Intuitively, in a biconnected component there is never a single bottleneck but always a backup path to reach any node. This ensures the function of the network even in case that an arbitrary edge has too low throughput or an arbitrary node of the biconnected component is damaged.

At the same time, the biconnected component is a simple concept enabling to some extent its treatment by analytical means for some graph ensembles. For example, the mean size $\langle S_2 \rangle$ of the biconnected component for a graph with a given degree distribution is known [18]. Also, the percolation transition of the biconnected component for scale-free and ER graphs is known to coincide with the percolation transition of the single connected component, and its finite size scaling behavior is known [22]. Nevertheless, a full description of any random variable is only obtained if its full probability distribution is known. To our knowledge, concerning the size of the biconnected component this has not been achieved so far for any graph ensemble, neither analytically nor numerically.

For few network observables and some graph ensembles results concerning the probability distributions have already been obtained so far. For the size of the connected component on ER random graphs analytical results [23] for the rate function exist, i.e., the behavior of the full distribution for large graph sizes N. Numerically it was shown that this is already for relatively small N a very good approximation [24]. Corresponding numerical results for two-dimensional percolation have been obtained as

---

[a] e-mail: hendrik.schawe@uni-oldenburg.de

well [24]. Similarly there are numerical, but no analytical works, scrutinizing the size of the related 2-core over most of its support again for ER random graphs [25].

Since similar results seem not to be available concerning the biconnected components, and given its importance for network robustness, this is an omission that we will start to cure with this study. Here, we numerically [26] obtain the probability density function of the size of the largest biconnected component over a large part of its support, i.e., down to probabilities smaller than 10^{-100}. This enables us also to directly observe large deviation properties, and shows strong hints that the large deviation principle holds [27,28] for this distribution.

The remainder of this manuscript gives definitions of the graph ensembles and the properties of interest, as well as some known results, in Section 2.1 and explains the sampling methods needed to explore the tails of the distributions in Section 2.2. The results of our simulations and a discussion will follow in Section 3. Section 4 summarizes the results.

2 Models and methods

2.1 Biconnected components of random graphs

A *graph* $G = (V, E)$ is a tuple of a set of nodes V and edges $E \subset V^{(2)}$. A pair of nodes i, j are called *connected*, if there exists a *path* of edges $\{i, i_1\}, \{i_1, i_2\}, .., \{i_{k-1}, i_k\}, \{i_k, j\}$ between them. A *cycle* is a closed path, i.e., the edge $\{i, j\}$ exists and i and j are connected in $G' = (V, E \setminus \{i, j\})$. The *connected components* are the maximal disjoint subgraphs, such that all nodes of each subgraph are connected.

A *biconnected component* (sometimes *bicomponent*) of an undirected graph is a subgraph, such that every node can be reached by two paths, which are distinct in every node except the start and end node. Thus, if any single node is removed from a biconnected component it will still be a connected component. Therefore clearly, each biconnected component is a connected component. We will also look shortly at *bi-edge-connected components*, which are very similar, but the two paths may share nodes as long as they do not share any edge. Note that a biconnected component is always a bi-edge-connected component, but the reverse is not necessarily true. An example is shown in Figure 1. In this study, we will study mainly the largest biconnected component G_{bi}. Note that, while every biconnected component is also a connected component, the largest biconnected component does not need to be a subgraph of the largest connected component G_{cc}, it may be part of another, smaller, connected component. However, its size $S_2 = |G_{\mathrm{bi}}|$ is always smaller or equal than the size of the largest connected component $S = |G_{\mathrm{cc}}|$. Similarly, the size $S_{2\text{-core}}$ of the largest connected component of the 2-*core*, the subgraphs that remain after iterative removal of all nodes with degree less than 2, is an upper bound on S_2, since the 2-core of a graph consists of bicomponents possibly linked by single edges. In Figure 1, the largest components of each type are visualized for an example connected graph. In fact for the sizes of the largest of the above introduced subgraphs, the following relation holds.

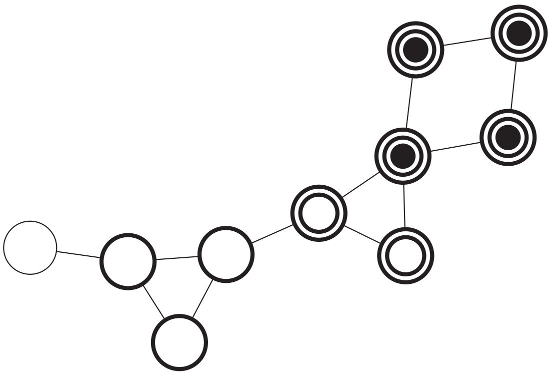


Fig. 1. Every node is part of the connected component, nine nodes with bold borderline are part of the 2-core, six nodes containing a circle are part of the largest bi-edge-connected component and all nodes containing a black dot are part of the largest biconnected component.

$$S \geq S_{2\text{-core}} \geq S_{2\text{-edge}} \geq S_2. \tag{1}$$

As we will see below, for the ensemble of ER random graphs in the percolating phase, the distributions of $S_{2\text{-core}}, S_{2\text{-edge}}$, and S_2 are actually very similar to each other. One has indeed to inspect the far tails of the distributions to see differences, which also justifies that we study the large-deviation properties here. For the ensemble of Barabási–Albert (BA) graphs we study, the same is true. While the distributions of $S_{2\text{-core}}$ and S_2 look very similar in the main region, a qualitative difference is observable in the tail of small components. The difference is even more pronounced than for the ER case, since the general form of the distribution changes qualitatively to a convex shape for $P(S_{2\text{-core}})$.

The classical way to find biconnected components of a graph [29] is based on a depth first search and thus runs in linear time. For each connected component a depth first search is started at an arbitrary root node of that component. For each node the current *depth* of the search, i.e., at which level in the tree traversed by the depth first search the node is located, and the *lowpoint* saved. The lowpoint is the minimum of the depth of the neighbors (in the graph) of all descendants of the node (in the tree). Iff the depth of a node is less or equal the lowpoint of one of its children (in the tree), this node separates two biconnected components and is called *articulation point*. For the root node of the search there is an exception. It is an articulation point, iff it has more than one child. The articulation points separate biconnected components and are members of all biconnected components separated by them. A better illustrated explanation can be found in reference [30]. After finding all biconnected components, we measure the size of the largest. We used the efficient implementation of this algorithm provided by the LEMON graph library [31].

The mean size of the biconnected component of graphs with a given degree distribution p_k is known for large graphs [18,32] to be

$$\langle S_2 \rangle = 1 - G_0(u) - (1-u)G_0'(u), \tag{2}$$

where $G_0(z) = \sum_k p_k z^k$ is the probability generating function, G_0' its derivative, and u the probability to reach a node not part of the giant connected component when following an edge. u is determined by the solution of

$$u = \sum_{k=0}^{\infty} q_k u^k, \tag{3}$$

with the excess degree distribution $q_k = (k+1)p_{k+1}/\langle k \rangle$. Knowing the degree distribution of ER graphs $G(N,p)$ to be

$$p_k = \binom{N-1}{k} p^k (1-p)^{N-1-k}, \tag{4}$$

allows the numerical evaluation of equation (3). We will compare these predictions to our simulational results to scrutinize the behavior for finite N.

The ensemble of *ER* graphs $G(N,p)$ consists of N nodes and each of the $N(N-1)/2$ possible edges occurs with probability p. The connectivity $c = Np$ is the average number of incident edges per node, the average *degree*. At $c_c = 1$ this ensemble shows a *percolation transition*. That is in the limit of large graph sizes N the size of the largest connected component is of order $\mathcal{O}(N)$ above this threshold and of order $\mathcal{O}(1)$ below. Interestingly this point is also the percolation transition of the biconnected component [22].

To a lesser extent we also study *BA* graphs [15]. The ensemble of BA graphs is characterized by a tunable mean degree $\langle k \rangle$ and its degree distribution follows a powerlaw $p(k) \propto k^{-3}$. Realizations are constructed using a growth process. Starting from a fully connected subgraph of m_0 (here $m_0 = 3$) nodes, in every iteration one more node is added and connected to $m \leq m_0$ existing nodes j with a probability $p_j \propto k_j$ dependent on their degree k_j until the size of the graph is N. The parameter $2m = \langle k \rangle$ by construction. Since $m = 1$ will always result in a tree, which is not biconnected at all and $m \geq 2$ will always be a full biconnected component, we will allow fractional $1 < m < 2$ in the sense, that one edge is always added and a second with probability $m - 1$.

We also take a brief look at the *configuration model* (CM) [33], an ensemble of graphs constructed to follow an arbitrary degree distribution. To sample the space of all simple graphs, i.e., graphs without self-loops and multi-edges, of the CM, one has to generate first a random degree distribution, add stubs to the nodes according to the degree distribution, connect the stubs randomly, and start from scratch in the case that the result is not a simple graph [34]. This means the amount of random numbers needed to generate one instance of a CM graph will vary, sometimes very strongly, if the instances are "difficult" to construct.

2.2 Sampling

Since we are interested in the far tail behavior of the distribution of the size of the largest biconnected component, it is infeasible to use naive simple sampling, i.e., uniformly generating configurations, measuring the observable, and constructing a histogram. Instead we also use a Markov chain Monte Carlo-based importance sampling scheme to collect good statistics in the far tails. This technique was already applied to obtain the distributions over a large range for the score of sequence alignments [35–37], to obtain statistics of the convex hulls of a wide range of types of random walks [38–41], to work distributions for non-equilibrium systems [42], and especially to different properties of ER random graphs [24,25,43,44].

The Markov chain in this case is a chain of random number vectors $\boldsymbol{\xi}_t$, $t = 1, 2, \ldots$ Each entry of $\boldsymbol{\xi}_t$ is drawn from a uniform $U(0,1)$ distribution. Each vector serves as an input for a function which generates a random graph. Since all randomness is included in $\boldsymbol{\xi}_t$, the generated graph $G_t = G(\boldsymbol{\xi}_t)$ depends deterministically on $\boldsymbol{\xi}_t$. In this way, the Markov chain $\{\boldsymbol{\xi}_t\}$ corresponds to a Markov chain $\{G_t\}$ of graphs. This approach, of separating the randomness from the actually generated objects, has the advantage that for the Markov chain we can generate graph realizations of arbitrary ensembles from scratch, without having to invent a valid Markov chain change move for each ensemble. However, for ER graphs, we use a specialized change move for performance reasons. One change move is to select a random node i, delete all incident edges, and add every edge $\{i,j\}$ with $j \in V \setminus \{i\}$ with probability p. For the BA graphs such a simple change move is not trivial to construct. Therefore, for this type, we perform the typical growth process from scratch after changing one of the underlying random numbers in $\boldsymbol{\xi}_t$.

The main idea to obtain good statistics over a large part of the support, especially for probabilities smaller than, say, 10^{-100}, is to bias the generated samples toward those regions. Therefore, we will use classical Metropolis sampling to gather realizations of graphs G. The Markov chain underlying this method consists of either graph realizations G (ER case) or random number vectors $\boldsymbol{\xi}$ from which a graph realization can be constructed $G(\boldsymbol{\xi})$ (BA case). We will describe the process for the latter more general case. We start our Markov chain with some random state $\boldsymbol{\xi}_1$ and at every iteration we propose a new state $\boldsymbol{\xi}'$, i.e., replace a single entry of $\boldsymbol{\xi}_t$ with a new uniform random number, and generate a new realization $G(\boldsymbol{\xi}')$ from these random numbers. We will accept this proposal as the new state $\boldsymbol{\xi}_{i+1}$, with the classical Metropolis acceptance probability $p_{\text{acc}} = \min\{1, \mathrm{e}^{-\Delta S/T}\}$. This process is sketched in Figure 2. Since we are interested in the size of the largest biconnected component S, we will treat this observable as the "energy" of the realization. Thus, ΔS is the difference in energy between the old and proposed state. Otherwise the proposal is rejected, i.e., $\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_t$. Following this protocol, the Markov chain will equilibrate eventually and from thereon yield realizations $G(\boldsymbol{\xi})$ which are Boltzmann distributed with respect to some "artificial

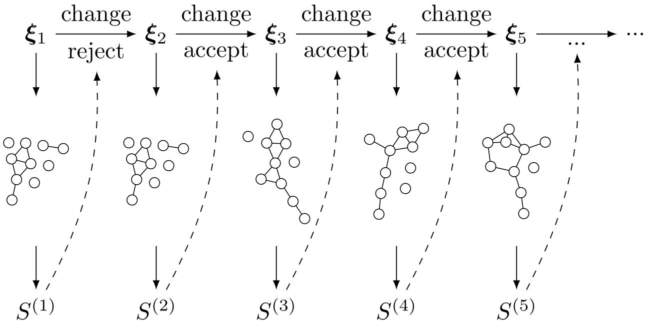


Fig. 2. Four steps of our importance sampling scheme at a small negative temperature, biasing toward a larger biconnected component.

temperature" T

$$Q_T(G) = \frac{1}{Z_T}\, \mathrm{e}^{-S(G)/T}\, Q(G), \tag{5}$$

where $Q(G)$ is the natural distribution of the realizations and Z_T the partition function, i.e., a normalization constant. Now, we can use the temperature T as a tuning parameter to adjust the part of the distribution we want to gather samples from. Low positive temperatures will bias the "energy" S toward smaller values because decreases in S are always accepted and increases in S are more often rejected. For negative T this bias works in the opposite way toward larger values of S, i.e., larger biconnected components in this case.

Note that, while this scheme is generally applicable to any model, there are models which are infeasible to equilibrate. As an example take the CM. The above described construction poses the problem that we use two types of random numbers. The first N random numbers to generate a degree distribution p_k and the remainder $\sum_{k=1}^{N-1} kp_k$ for the connections between the stubs. Thus, the amount of random number varies somehow. But this is true only if in the first attempt a valid set of edges is created. If not, one would need to perform for the current degree sequence one or several other attempts, creating the need for many more random numbers. Thus, a state ξ of the random numbers would be much larger, containing many numbers "in stock", much larger than needed to construct a typical graph which requires only one attempt. First, this makes in somehow numerically demanding. But even worse, in total this means that a small change to one of the first N random numbers typically leads to a strong change of the resulting graph, such that almost all changes of this kind will be rejected when approaching the tails. It might, of course, be possible to devise an efficient change move. Nevertheless, we were not able to sample the biconnected component of the CM in the far tails, and will only use data obtained by simple sampling for some qualitative comparisons of the three graph ensembles.

For any chosen temperature, the sampling will be restricted to some interval determined by the value of T. Thus, to obtain the desired distribution $P(S)$ over a large range of the support, simulations for many different temperatures have to be performed. We have to choose the temperatures T in a certain way, to be able to reconstruct the wanted distribution $P(S)$ from this data. First, we can transform $Q(G)$ into $P(S)$ by summing all realizations G, which have the same S. Hence we obtain with equation (5)

$$\begin{aligned} P_T(S) &= \sum_{\{G|S(G)=S\}} Q_T(G) \\ &= \sum_{\{G|S(G)=S\}} \frac{\exp(-S/T)}{Z_T} Q(G) \\ &= \frac{\exp(-S/T)}{Z_T} P(S). \end{aligned}$$

With this relation we can calculate the wanted, unbiased distribution $P(S)$ from measurements of our biased distributions $P_T(S)$. The ratios of all constants Z_T can be obtained by enforcing continuity of the distribution $P(S)$, i.e.,

$$P_{T_j}(S)\, \mathrm{e}^{S/T_j}\, Z_{T_j} = P_{T_i}(S)\, \mathrm{e}^{S/T_i}\, Z_{T_i}.$$

This requires that our measurements for $P_T(S)$ are at least pairwise overlapping such that there is no unsampled region between sampled regions. From pairwise overlaps the pairwise ratios Z_{T_i}/Z_{T_j} can be approximated. The absolute value of the Z_T can afterwards be obtained by the normalization of $P(S)$. Although the size of the largest biconnected component S_2 is a discrete variable for every finite N and should therefore be normalized such that the probabilities of every event should sum to one, i.e., $\sum_{i=0}^{N} p(S_2 = i/N) = 1$, we are mainly interested in the large N behavior and especially the rate function. This limit is continuous and should therefore be treated with a different normalization $\int_0^1 p(S_2)\, \mathrm{d}S_2 = 1$, which we approximate for finite N by the trapezoidal rule. Anyway, the difference here is just a factor N.

While this technique does usually work quite well and all distributions but one exception are obtained with this method, there are sometimes first order phase transitions within the finite temperature ensemble, rendering it infeasible, or at least very tedious, to acquire values inbetween two temperatures. This was a problem here for the modified BA graph at the largest simulated graph size N. This phenomenon is well known and explored in detail in [24]. We filled this gap by modified Wang–Landau simulations [45–49] with subsequent entropic sampling [50,51].

3 Results

We applied the temperature-based sampling scheme to ER with finite connectivities of $c \in \{0.5, 1, 2\}$ and BA with $m = 1.3$ over practically the whole support $S_2 \in [0,1]$ using around a dozen different temperatures for each ensemble and Markov chains of length $10^6 N$ to gather enough samples after equilibration and discarding correlated samples. Additionally for BA the range $S_2 \in [0.1, 0.35]$ was sampled using Wang–Landau's method and merged into results obtained from the temperature-based sampling for the remainder of the distribution. All error

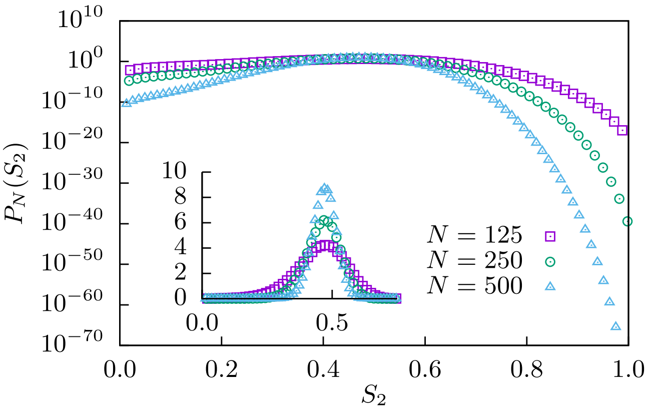


Fig. 3. Distributions of the size of the largest biconnected component S_2 for ER graphs at connectivity $c = 2$ and three different graph sizes N. The main plot shows the distributions in logarithmic scale to display the tails, the inset shows the same distributions in linear scale, where a concentration around the mean value $\langle S_2 \rangle$ (cf. Fig. 4) is visible. Note that despite $P_N(S_2)$ being a discrete distribution, it is normalized like a continuous distribution (see second to last paragraph of Sect. 2 for the rationale). (For clarity not every bin is visualized.)

estimates for the distributions are obtained via bootstrap resampling [52,53] but are always smaller than the symbol size and therefore not shown. Error estimates for fit parameters are Gnuplot's asymptotic standard errors corrected according to reference [53].

Examples for the distribution of the largest biconnected component's size for ER graphs at $c = 2$ are depicted in Figure 3 at three different graph sizes N. The inset shows the distribution in linear scale, where a concentration with increasing size N around the mean value is visible. While the main part of the distribution in the inset looks rather symmetric, the tails are obviously not. Also it is visible that the tails of the distribution get more and more suppressed when increasing the value of N.

Since the mean size of the biconnected component of ER is known for large enough graphs, we will compare the mean sizes of our simulations to the analytical expectation. Those results are shown in Figure 4, notice the broken $\langle S_2 \rangle$-axis. Apparently at $c = 2$ for small sizes N the analytical approximation, while close to our measurements, overestimates the size of the biconnected component slightly but the relative error diminishes for larger sizes. In fact, we extrapolated our measurements to the limit of large N using a power-law ansatz $\langle S_2 \rangle = aN^b + S_2^\infty$ yielding for $c = 0.5$ an offset S_2^∞ compatible within errorbars with the expectation $\langle S_2 \rangle = 0$ (exact values in the caption of Fig. 4), which is quite remarkable for our ad hoc fit function. The case $c = 1$ suggests a negative S_2^∞ close to zero, which is probably caused by correction to our assumed scaling law. The case $c = 2$ seems to converge to the limit of the analytical expectation also.

To compare the sizes of different relevant types of components, Figure 5 shows the distributions of the relative size of the largest connected component S [24], the largest 2-core $S_{\text{2-core}}$ [25], the largest bi-edge-connected component $S_{\text{2-edge}}$, and the largest biconnected component S_2 for $N = 500$ and $c = 2$ ER graphs. Interestingly, the distributions $P_N(S_2)$, $P_N(S_{\text{2-edge}})$, and $P_N(S_{\text{2-core}})$ are almost identical and only deviate in the region of very small components from each other. As would be expected by the order of equation (1), the probability to find very small 2-cores is lower than to find bi-edge-connected components of the same small size, which are again slightly less probable than biconnected components of that size. Anyway, when considering ER graphs, which exhibit by construction no particular structure, the robustness properties which are determined by the biconnected component, can be with very high probability readily inferred from the 2-core.

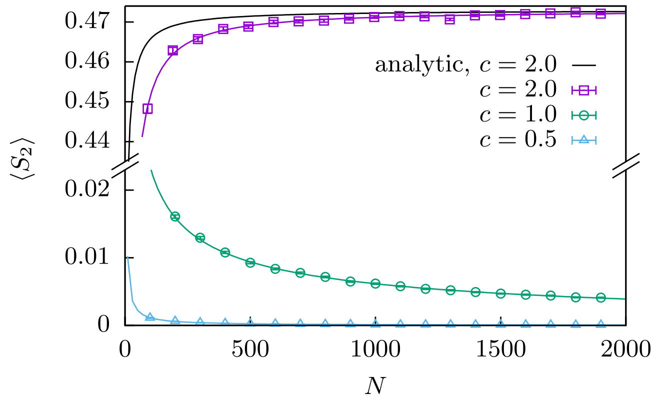


Fig. 4. Mean size of the largest biconnected component $\langle S_2 \rangle$ for different graph sizes N. Notice the broken $\langle S_2 \rangle$-axis. The black line denotes the analytic expectation for $c = 2$ from equation (2) [18]. The expectation for $c \le 1$ is $\langle S_2 \rangle = 0$. Fits to a power law with offset $\langle S_2 \rangle = aN^b + S_2^\infty$ lead to $S_2^\infty = -6(8) \times 10^{-6}$ for $c = 0.5$, $S_2^\infty = -0.0013(4)$ for $c = 1$, and $S_2^\infty = 0.4729(3)$ for $c = 2$.

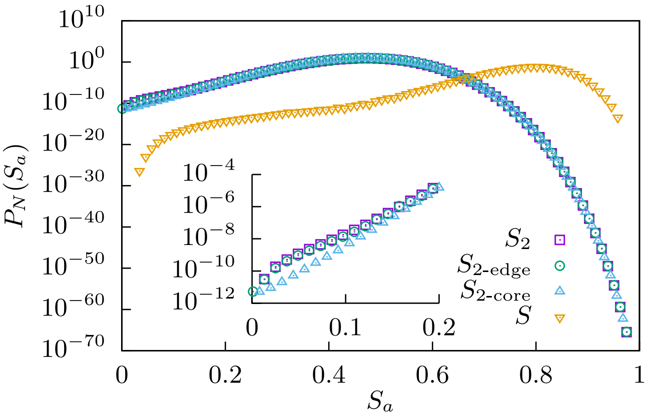


Fig. 5. Comparison of the relative sizes S_a, which can be any of the largest connected component S [24], the largest 2-core $S_{\text{2-core}}$ [25], the largest bi-edge-connected component $S_{\text{2-edge}}$, and the largest biconnected component S_2 for $N = 500$ and $c = 2$ ER graphs. The last three are nearly identical for sizes $S_a \gtrsim 0.2$. The inset shows a zoom to the very small components, which is the only region, where the three last types deviate considerably from one another. For clarity not every data point is visualized.

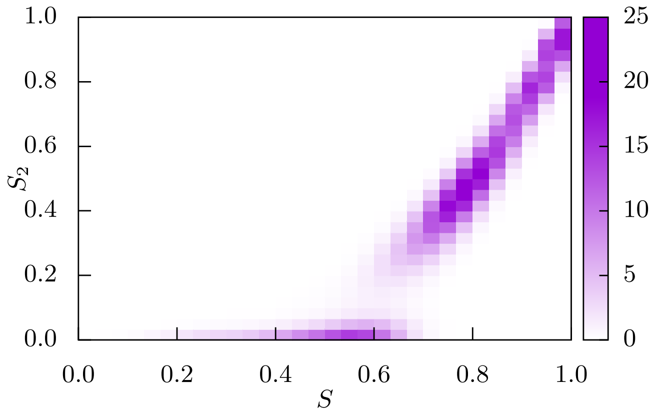


Fig. 6. Correlation histogram of our raw and biased simulational data for the $c = 2$ ensemble of ER graphs. A large biconnected component does most probably appear in graphs whose connected component is larger than $S \gtrsim 0.6$. This is a qualitative correlation plot only, as the exact values of the bins are dependent on the temperatures used for the simulation. The colorbar encodes a normalized probability density to encounter the corresponding pair of values in our finite-temperature ensembles.

To understand the reason for the sizes S_2, $S_{2\text{-edge}}$, and $S_{2\text{-core}}$ to be so similar for graphs with independently created edges like ER and CM, consider the argumentation of reference [18], where an upper bound equation (2), which becomes exact for $N \to \infty$, is derived as the probability of a node to have two edges connecting to the giant component. For finite graphs this is just an upper bound, as two paths to the giant component are necessary for a node to be part of the giant biconnected component but not sufficient. To be sufficient, we have to ensure that the two paths do not share any nodes (or any edges for the bi-edge-connected component). Similarly, this criterion also works for the giant 2-core in the limit of large N, but here it does not provide an upper bound, because two independent paths are not necessary. For a node to not be a leaf at some point in the process of finding a 2-core, it has to be connected with two edges to other biconnected components, possibly the same. To be part of the giant two core it has to be in the connected component. Using the argumentation of reference [18] that small biconnected components are very rare for large values of N, we see that the two biconnected components with which any node of the giant 2-core is connected are with high probability just part of the giant biconnected component. One would therefore expect that in the $N \to \infty$ case these observables behave the same. And indeed, in our data we observe that the regime where they behave most differently is in graphs with atypically small sizes of the components, again highlighting the power of the large-deviation approach without which it would be impossible to observe these differences. Keep in mind that this argument does only work for models with independently placed edges, like the ER or CM models. Indeed we will see below that for the BA model, the differences between the component types are much larger.

To understand the topology of the instances of very low probabilities better, we will look at the correlations of the size of the largest connected component S and the largest biconnected component in Figure 6. Note that this histogram does not reflect the probabilities, but does count the instances we generated within one of our simulations, i.e., data for many different temperatures are shown without correction for the introduced bias. Anyway, it is instructive to look at this sketch for qualitative understanding. This data is for $c = 2$ ER at $N = 500$. We observe that, even for our biased sampling, there are basically no large biconnected components if the connected component is smaller than $S \lesssim 0.6$. Above this point, we observe larger biconnected components, but generally very few around the size $S_2 \approx 0.2$. Above $S \gtrsim 0.6$ the size of the largest biconnected component is strongly correlated with the size of the largest connected component.

For a qualitative understanding of this behavior, consider the following heuristic argument. For the instances without or with very small biconnected components, i.e., only short cycles, the graph is basically tree-like. Larger biconnected components are then created by connecting two nodes of the tree with each other, leading to a cycle which is on average in the order of the size of the tree, leading to the jump in the size of biconnected components. The configurations with smaller biconnected components are apparently entropically suppressed.

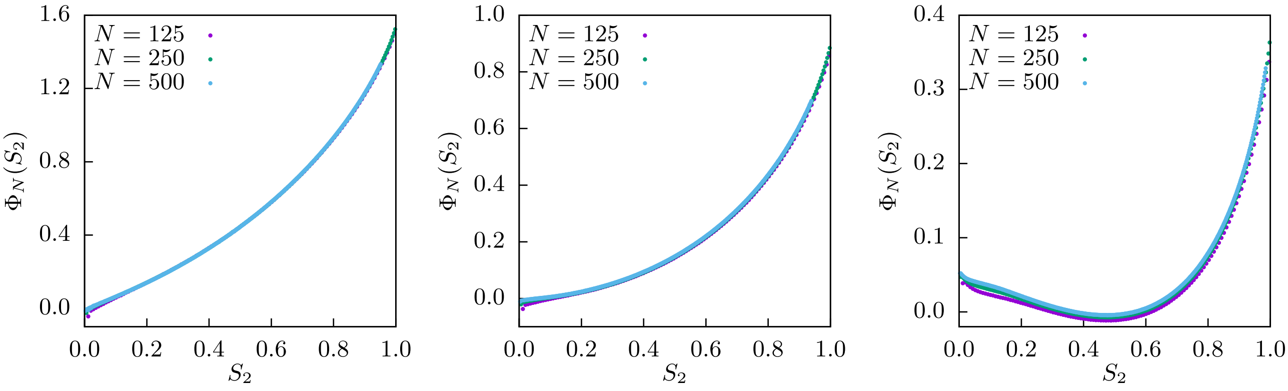
Next we will look at the empirical large deviation *rate function* of the measured distributions. The rate function Φ describes the behavior of distributions, whose probability density decays exponentially in the tails in respect to some parameter N. In this case, the parameter N is the graph size. For increasing graph size N the biconnected components which are not typical will be exponentially suppressed. To be more precise, the definition of the rate function $\Phi(S_2) \geq 0$ is via $P_N(S_2) = \mathrm{e}^{-N\Phi(S_2)+o(N)}$ for the large N limit with the Landau symbol $o(N)$ for terms of order less than N. If a rate function exists, we can read off, for example, that the value S_2^* at which the rate function $\Phi(S_2^*) = 0$ is the value around which the probability concentrates, i.e., the size of the biconnected component is self averaging.

Since we obtained the distributions P_N over most of their support but at finite N, we can only access the *empirical rate function* Φ_N for finite values of N, i.e.,

$$\Phi_N(S_2) = -1/N \log P_N(S_2) + o(N)/N. \qquad (6)$$

Note also that the empirical rate functions do contain all information of the measured distributions P_N, such that we will only visualize either P_N or Φ_N in the following. Note also that the $o(N)$ term leaves enough freedom at finite N to shift the empirical rate function a bit such that negative values can occur. However, if for increasing values of N a convergence to a limiting curve is visible, this limiting curve is the actual rate function and one says the distribution follows the *large deviation principle* [27,28].

In Figure 7 the empirical rate functions for ER at different connectivities c and for different sizes N are shown. The data of the distribution has a very high precision as the values of the rate function Φ_{500} in Figure 7a reaches values of $\Phi_{500} = 1.5$ corresponding to probabilities less than 10^{-300}. Already these comparatively small values



(a) $c = 0.5$

(b) $c = 1$

(c) $c = 2$

Fig. 7. Empirical rate function $\Phi_N(S_2)$ for multiple graph sizes N and connectivities c of the ER graph ensemble; (a) below the percolation transition, (b) at the percolation transition and (c) above the percolation transition. For the cases (a) and (b) finite size effects are minor and a very fast convergence towards the rate function valid for large values of N is visible. The case (c) shows a qualitatively different behavior, where the minimum of the empirical rate function is shifted to finite values. The convergence towards the actual rate function is visible.

of N show remarkably similar empiric rate functions and strongly hint at a convergence to a limit form. Also, while near the minima negative values of the empirical rate function occur, it is clear that the convergence is toward zero at those positions, such that it is plausible that the actual rate function is non-negative. While the empirical rate functions in the right tail for larger than typical components S_2 are already almost indistinguishable, the convergence seems a bit slower in the left tail of smaller than typical components. This behavior is very similar to the behavior of the sizes of the connected component [24] and the 2-core [25]. This means that, the large deviation principle seems to hold for this distribution.

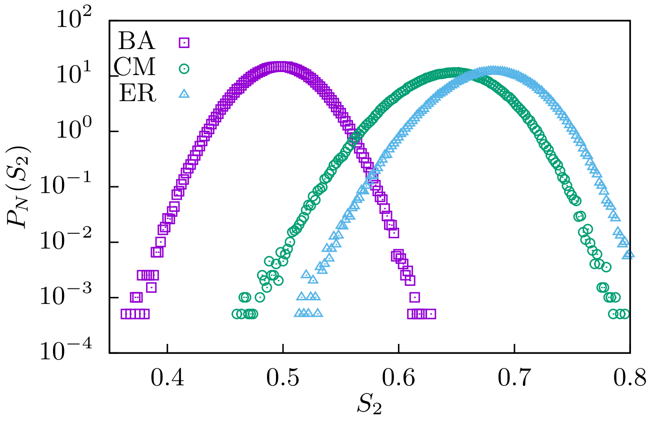


Fig. 8. Distribution $P_N(S_2)$ of the size of the giant biconnected component in the BA model with a mean degree of $\langle k\rangle = 2.6$ and a power-law degree distribution $p_k \sim k^{-3}$ in comparison to the CM with approximately the same mean degree and the same exponent governing the power-law degree distribution and the ER model with the same mean degree. Apparently the BA has a far smaller biconnected component than the other two, but is connected. Samples were taken for graphs of size $N = 500$.

Let us now take a look at the BA model. First, we will just compare the mean size of the biconnected component $\langle S_2\rangle$, which we measured on the BA model similar to the extrapolation in Figure 4 ($300 \leq N \leq 1000$), resulting in $\langle S_2^{\mathrm{BA}}\rangle \approx 0.4982(4)$ for large values of N. In comparison to the analytical value $\langle S_2^{\mathrm{ER}}\rangle = 0.6811\ldots$ for large N [18] for the ER graph with the same mean degree $\langle k\rangle = 2.6$, we observe that the connectedness and robustness against random failures of the BA have to be paid with a decreased robustness against worst case failures, which is a phenomenon observed before [16]. To check whether this effect is caused by the degree distribution or the correlations, we look at the CM with a similar degree distribution, namely a Pareto distribution $p(k) = 2k_0k^{-3}$ with the same exponent as for the BA model and with a tunable minimum k_0, which we change to result in a mean degree of $\langle k\rangle \approx 2.6$. Note that due to the discrete nature of the degree distribution this does not result in a perfect power law, in particular $p(\lfloor k_0\rfloor)$ is lower than for a perfect power law, but it should be close enough for our purposes. Here, we observe $\langle S_2^{\mathrm{CM}}\rangle \approx 0.6650(6)$. This is slightly lower than the value for ER. This small difference between the ER and CM model is also visible when looking at the typical region of the distribution of S_2 as shown in Figure 8. This indicates that hubs are a cause to destabilize a network against worst-case failures, but considerably a smaller cause than for BA, indicating that the correlations and the forced connectedness of BA graphs lead to less redundancy in the network. Thus, since the behavior of CM and ER is quite similar, and because of the algorithmic complexity we encounter for the CM model, as mentioned above, we proceed with the large-deviation behavior of the BA model.

The empirical rate function of the largest biconnected component of the BA ensemble at $m = 1.3$, i.e., a mean degree of $\langle k\rangle = 2.6$, is shown in Figure 9. The empirical rate function and therefore the distribution does look qualitatively similar to the $c = 2$ case of the ER ensemble (cf. Fig. 7). The dip around $S_2 \approx 0.2$ is more pronounced leading to a more severe discontinuity in the

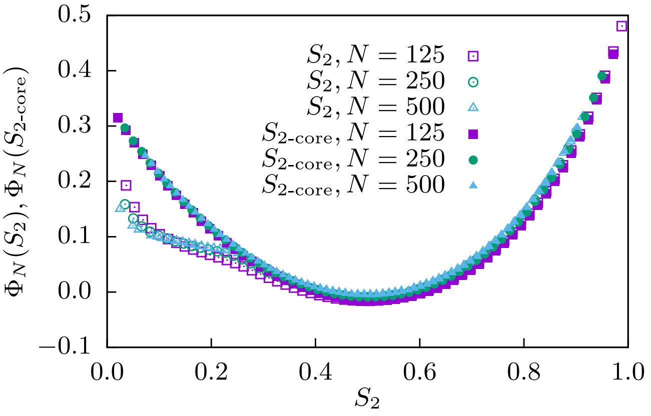


Fig. 9. Empirical rate function $\Phi_N(S_2)$ for multiple graph sizes N of the BA graph ensemble with $m = 1.3$. A convergence toward an asymptotic form for large values of N is visible. For comparison also the empirical rate function $\Phi_N(S_2)$ characterizing the distribution of the 2-core is shown. It deviates qualitatively in the far left tail.

simulated finite temperature ensemble necessitating the use of Wang–Landau sampling. The main qualitative difference of the behavior of the two distributions is visible in the small sizes of the largest biconnected components S_2, where the empirical rate functions cross each other, hinting at some kind of finite size effect suppressing very small biconnected components in small graphs.

In comparison with the empirical rate function of the size of the 2-core $S_{2\text{-core}}$, also shown in Figure 9, their difference in the region of very small bicomponents respectively 2-cores, which was already observable in ER, is very strong in the BA ensemble. Despite those two observables being almost indistinguishable in the main region, they show strongly different behavior in their overall shape, i.e., the distribution of the 2-core seems convex over the region we obtained statistics for. Note, however, that in contrast to the ER or even CM with the same degree distribution, we do not expect the two distributions to be indistinguishable in the $N \to \infty$ limit. While the argumentation for the ER needed the prerequisite of independent edges, the BA shows strong correlations, e.g., cliques of 3 nodes occur more often than in the ER or a CM with the same degree distribution. As numerical evidence that the two distributions do not become identical in the $N \to \infty$ limit, observe in Figure 9 that while the 2-core is almost converged already, the left tail of $\Phi_N(S_2)$ tends away with increasing graph sizes N. Thus, for infinite graph size, the difference between S_2 and $S_{2\text{-core}}$ will remain strong and extensive in the left tail.

4 Conclusions

The biconnected component is a fundamental observable of any graph related to its robustness. In general, we identified competing properties, which influence the robustness of networks, e.g., while the specific growth process for BA graphs leads to a large connected component, it also leads to a smaller biconnected component. Tests on the CM show that the degree distribution has an influence on the size of the biconnected component, but the construction rules of BA graphs have a larger impact. This supports that networks originating from preferential attachment processes might be particularly susceptible to targeted attacks or worst-case failures as compared to networks following the same degree distribution but exhibiting independently drawn edges.

On a more fundamental level, the distribution of its size has not been studied before, to our knowledge. We used sophisticated sampling methods to obtain the distributions of the size of the largest biconnected component S_2, for multiple ER graph ensembles and a modified BA graph ensemble, over a large part of their support. For the ER ensemble, looking into the large deviation tails of this distribution shows qualitative differences between the size of the 2-core and the biconnected component, which are otherwise not well observable and which we expect to vanish for large systems. Even more interesting is the case for the BA ensemble where the overall shape of the distributions seems to differ also for large systems. While the 2-core distribution seems convex, the distribution of the biconnected component shows a “shoulder”. These qualitative differences, however, are only apparent below probabilities of 10^{-20} and are therefore unobservable using conventional methods.

Further, the empirical rate functions are already for the small sizes that we simulated very close to each other hinting at a very fast convergence to the limiting form. Thus, our results indicate that the large deviation principle holds for the numerically obtained distributions. This “well-behaving” of our numerical results may make it promising to address the distribution of the biconnected component by analytical means, which has not been done so far to our knowledge. Furthermore, it would be interesting to study other network ensembles, which are even more relevant for modeling robustness properties, e.g., two-dimensional networks modeling power grids [8] and other transportation networks.

This work was supported by the German Science Foundation (DFG) through the grant HA 3169/8-1. We also thank the GWDG (Göttingen) for providing computational resources.

Author contribution statement

AKH conceived the study, HS wrote the first draft of the manuscript and generated most of the new data. All authors contributed ideas, simulation data, and analysis to this study. All authors were involved in the preparation of the manuscript.

References

1. M.E.J. Newman, SIAM Rev. **45**, 167 (2003)
2. S.N. Dorogovtsev, J.F.F. Mendes, *Evolution of Networks: From Biological Nets to the Internet and WWW* (Oxford University Press, Oxford, 2006)
3. M. Newman, A.L. Barabási, D. Watts, *The Structure and Dynamics of Networks* (Princeton University Press, 2006)

4. M. Newman, *Networks: an Introduction* (Oxford University Press, Princeton, 2010)
5. A. Barrat, M. Barthélemy, A. Vespignani, *Dynamical Processes on Complex Networks* (Cambridge University Press, Cambridge, 2012)
6. M.L. Sachtjen, B.A. Carreras, V.E. Lynch, Phys. Rev. E **61**, 4877 (2000)
7. M. Rohden, A. Sorge, M. Timme, D. Witthaut, Phys. Rev. Lett. **109**, 064101 (2012)
8. T. Dewenter, A.K. Hartmann, New J. Phys. **17**, 015005 (2015)
9. R. Cohen, K. Erez, D. ben Avraham, S. Havlin, Phys. Rev. Lett. **85**, 4626 (2000)
10. D.S. Lee, H. Rieger, EPL (Europhys. Lett.) **73**, 471 (2006)
11. C.M. Ghim, K.I. Goh, B. Kahng, J. Theor. Biol. **237**, 401 (2005)
12. P. Kim, D.S. Lee, B. Kahng, Sci. Rep. **5**, 15567 (2015)
13. P. Erdős, A. Rényi, Publ. Math. Inst. Hungar. Acad. Sci. **5**, 17 (1960)
14. D.J. Watts, S.H. Strogatz, Nature **393**, 440 (1998)
15. A.L. Barabási, R. Albert, Science **286**, 509 (1999)
16. R. Albert, H. Jeong, A.L. Barabási, Nature **406**, 378 (2000)
17. D.S. Callaway, M.E.J. Newman, S.H. Strogatz, D.J. Watts, Phys. Rev. Lett. **85**, 5468 (2000)
18. M.E.J. Newman, G. Ghoshal, Phys. Rev. Lett. **100**, 138701 (2008)
19. C. Norrenbrock, O. Melchert, A.K. Hartmann, Phys. Rev. E **94**, 062125 (2016)
20. G. Bianconi, Phys. Rev. E **97**, 022314 (2018)
21. G. Bianconi, Phys. Rev. E **96**, 012302 (2017)
22. P. Kim, D.S. Lee, B. Kahng, Phys. Rev. E **87**, 022804 (2013)
23. M. Biskup, L. Chayes, S.A. Smith, Random Struct. Algor. **31**, 354 (2007)
24. A.K. Hartmann, Eur. Phys. J. B **84**, 627 (2011)
25. A.K. Hartmann, Eur. Phys. J. Special Topics **226**, 567 (2017)
26. A.K. Hartmann, *Big Practical Guide to Computer Simulations* (World Scientific, Singapore, 2015)
27. F. den Hollander, *Large Deviations* (American Mathematical Society, Providence, 2000)
28. H. Touchette, Phys. Rep. **478**, 1 (2009)
29. J. Hopcroft, R. Tarjan, Commun. ACM **16**, 372 (1973)
30. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, *Introduction to Algorithms* (MIT Press, MA, 2009)
31. B. Dezső, A. Jüttner, P. Kovács, Electron. Notes Theor. Comput. Sci. **264**, 23 (2011) (Proceedings of the Second Workshop on Generative Technologies (WGT) 2010)
32. G. Ghoshal, Ph.D. thesis, University of Michigan, 2009
33. M.E.J. Newman, S.H. Strogatz, D.J. Watts, Phys. Rev. E **64**, 026118 (2001)
34. H. Klein-Hennig, A.K. Hartmann, Phys. Rev. E **85**, 026101 (2012)
35. A.K. Hartmann, Phys. Rev. E **65**, 056102 (2002)
36. S. Wolfsheimer, B. Burghardt, A.K. Hartmann, Algorithm. Mol. Biol. **2**, 9 (2007)
37. P. Fieth, A.K. Hartmann, Phys. Rev. E **94**, 022127 (2016)
38. G. Claussen, A.K. Hartmann, S.N. Majumdar, Phys. Rev. E **91**, 052104 (2015)
39. T. Dewenter, G. Claussen, A.K. Hartmann, S.N. Majumdar, Phys. Rev. E **94**, 052120 (2016)
40. H. Schawe, A.K. Hartmann, S.N. Majumdar, Phys. Rev. E **97**, 062159 (2018)
41. H. Schawe, A.K. Hartmann, arXiv:1808.10698 (2018)
42. A.K. Hartmann, Phys. Rev. E **89**, 052103 (2014)
43. A. Engel, R. Monasson, A.K. Hartmann, J. Stat. Phys. **117**, 387 (2004)
44. A.K. Hartmann, M. Mézard, Phys. Rev. E **97**, 032128 (2018)
45. F. Wang, D.P. Landau, Phys. Rev. Lett. **86**, 2050 (2001)
46. F. Wang, D.P. Landau, Phys. Rev. E **64**, 056101 (2001)
47. B.J. Schulz, K. Binder, M. Müller, D.P. Landau, Phys. Rev. E **67**, 067102 (2003)
48. R.E. Belardinelli, V.D. Pereyra, Phys. Rev. E **75**, 046701 (2007)
49. R.E. Belardinelli, V.D. Pereyra, J. Chem. Phys. **127**, 184105 (2007)
50. J. Lee, Phys. Rev. Lett. **71**, 211 (1993)
51. R. Dickman, A.G. Cunha-Netto, Phys. Rev. E **84**, 026701 (2011)
52. B. Efron, Ann. Stat. **7**, 1 (1979)
53. A.P. Young, *Everything You Wanted to Know About Data Analysis and Fitting but Were Afraid to Ask*, SpringerBriefs in Physics (Springer International Publishing, Switzerland, 2015)

B. Technical Details

B.1. Gluing the Single Distributions

This section describes the technical details of the procedure to transform the single corrected histograms at different temperatures into one continuous distribution. This is an extension to Section 2.2.4.

Given two overlapping distributions $P_{\Theta_{i-1}}$ and P_{Θ_i} the ratio of their free parameters $Z_{\Theta_{i-1}}/Z_{\Theta_i}$, or formulated in logarithms $\ln Z_{\Theta_{i-1}} - \ln Z_{\Theta_i}$, is obtained by enforcing equality over the overlap (see also Equation (2.30)).

$$Z_{\Theta_{i-1}} P_{\Theta_{i-1}}(S) \mathrm{e}^{S/\Theta_{i-1}} = Z_{\Theta_i} P_{\Theta_i}(S) \mathrm{e}^{S/\Theta_i}, \tag{B.1}$$

For simplicity, let $r_i = \ln\left(P_{\Theta_i}(S)\mathrm{e}^{S/\Theta_i}\right)$, which leads to

$$\ln Z_{\Theta_{i-1}} - \ln Z_{\Theta_i} = r_i - r_{i-1}. \tag{B.2}$$

Since the data we collect is subject to sampling errors and the overlap should be larger than a single bin, the equality can not be enforced strictly. Instead we minimize the difference while giving more weight[1] to samples with a smaller statistical error. Therefore we use the intuitive way[2] of weighting every bin j of histogram i with the number of samples n_j^i. Since we calculate the difference of two bins, we take the conservative choice to weight the difference with the weight of the bin having fewer samples. Thus the estimate for the ratio is

$$\ln Z_{\Theta_{i-1}} - \ln Z_{\Theta_i} = \frac{1}{\sum_j \min\left\{n_j^{i-1}, n_j^i\right\}} \sum_j \min\left\{n_j^{i-1}, n_j^i\right\} \left(r_i(S_j) - r_{i-1}(S_j)\right). \tag{B.3}$$

If two overlapping parts cannot be smoothly merged, it is often a hint that the equilibrium was not reached. At this stage one could introduce a formal criterion, for example if the merge leads for some fraction of data points in the overlap from the simulation at Θ_i to deviate more than, say, three times their standard deviation from the data points in the same bin of simulation Θ_{i-1}. In this case longer simulations and a more careful estimation of the equilibration time usually lead to better results. However, in the publications of this thesis equilibration times were usually very short,

---

[1] Note that Article A.2 does not utilize the weighting that is introduced in this section. Though, the unweighted variant will not lead to systematical errors and since the overlaps consist typically of many bins with good statistics and few with bad statistics, the results are very similar.

[2] and according to Reference [19, p. 219 ff] the "standard way". Note however that there are also different approaches, e.g., based on iterative optimization methods [168, p. 16 ff].

such that detection of not equilibrated simulations did not pose any problems. As soon as all ratios of adjacent "temperatures" are obtained, the single distributions are shifted accordingly. The unnormalized distribution $\widetilde{P}(S_j)$ over the whole covered range is then obtained by a weighted average over all partial, corrected and shifted, but unnormalized distributions. Also this step is performed on the logarithms of the collected data, i.e., the averaging is again a geometric mean

$$\ln \widetilde{P}(S_j) = \ln \left(P(S_j) Z_1\right) = \frac{1}{\sum_i n_j^i} \sum_i n_j^i \ln P_{\Theta_i}(S_j) + S_j/\Theta_i + \sum_{k=1}^{i} \ln Z_{\Theta_k}. \tag{B.4}$$

The absolute value of the normalization constants, e.g., the first, Z_1 can be obtained by using that $\int_{-\infty}^{\infty} P(S)\,\mathrm{d}S = 1$. We can thus obtain Z_1 numerically by performing an integration, e.g., with the trapezoidal rule for equal-width bins

$$\frac{1}{Z_1} = \frac{b-a}{2N} \sum_{k=1}^{N} \left[\widetilde{P}(S_{k-1}) + \widetilde{P}(S_k)\right], \tag{B.5}$$

where a and b are the lowest and highest bin positions, N the number of bins and S_k the position of the k-th bin. As a technicality, we subtract the maximum of $\ln \widetilde{P}(S_j)$ before performing the exponentiation to avoid difficulties with the numerical precision of our datatypes.

B.2. Further Convex Hull Algorithms

This section will introduce an heuristic to reduce the run time of any exact algorithm and two further exact algorithms for the evaluation of the convex hull, which were candidates for the simulation program. After they are introduced in Appendices B.2.1 to B.2.3, the relative performance of all considered algorithms is shortly analyzed in Appendix B.2.4, which is the basis of the decision which algorithm to use for our studies.

B.2.1. Akl-Toussaint Elimination Heuristic

The *Akl-Toussaint heuristic* [169] is a fast way to discard points which can not be part of the convex hull and can be used to preprocess the point set before applying one of the exact algorithms. For brevity sake, only the $d = 2$ special case is explained as the generalization to higher dimensions is straight forward. Since every point inside the convex hull of a subset of the points, is also in the convex hull of all points, one does not need to consider these points during the calculation of the actual convex hull.

The easiest way to create a polygon which encloses many points, is to choose the 4 points with maximum, respectively minimum values in x and y direction as shown in Figure B.1(a). In fact, Reference [170] proposes that it is most efficient to choose 4 additional points: the maximum and minimum of $x + y$ and $x - y$. In Figure B.1(b) the improvement is visualized with an example. Usually a considerable amount of

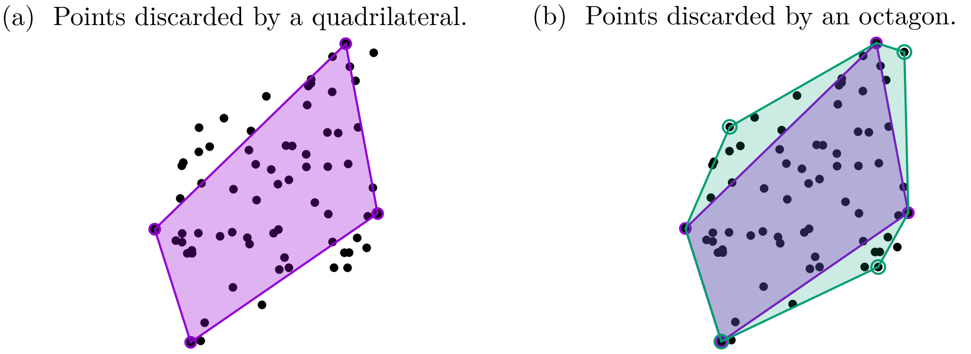

(a) Points discarded by a quadrilateral. (b) Points discarded by an octagon.

Figure B.1.: Example for the Akl-Toussaint elimination heuristic. (a) The simple method with 4 extreme points spanning a quadrilateral. (b) The variation using 8 points (but only 7 distinct in this example) to span a polygon.

points is inside this polygon and can thus be discarded. Since the operation of testing whether a point is inside a polygon (cf. Section 3.3.2) is quite cheap, this simple $\mathcal{O}(n)$ technique often leads to considerable speedups, especially for large point sets.

While this method can easily be extended to higher dimensions, it is less efficient for higher dimensions. Also note, that the Quickhull algorithm implicitly performs this heuristic, but with a triangle in $d = 2$ and a tetrahedron in $d = 3$, thus no large improvements in combination with the Quickhull are to be expected.

B.2.2. Jarvis' March / Gift Wrapping

Jarvis' march [171] is an *output dependent* convex hull algorithm, which means that its time complexity in two dimensions $\mathcal{O}(nh)$ depends on the number of points that are part of the convex hull h. In the worst case, e.g., points on a circle, all points are part of the convex hull and therefore the complexity would be $\mathcal{O}(n^2)$.

The main idea is to start at one point known to be a vertex of the convex hull, e.g., the point with minimum x-coordinate, and turning a hyperplane around this point until it hits $d - 1$ additional points, which become part of the hull. This is repeated for every vertex hit by the facet and not already part of the hull. Since this is basically the same method humans use to wrap gifts, it is also known as *Gift Wrapping*.

For an efficient implementation, here for the example in $d = 2$, it can be interpreted as maximizing the angle between the last found facet $(\boldsymbol{p}_{i-1}, \boldsymbol{p}_i)$ and any point $\boldsymbol{q}'$, i.e.,

$$\boldsymbol{q} = \arg\max_{\boldsymbol{q}'} \left\{ \measuredangle(\boldsymbol{p}_{i-1}, \boldsymbol{p}_i, \boldsymbol{q}') \right\}.$$

The next facet is then $(\boldsymbol{p}_i, \boldsymbol{q})$ and the process is iterated until the start is reached, i.e., $\boldsymbol{q} = \boldsymbol{p}_0$. Since maximizing this angle is equivalent to finding the point $\boldsymbol{q}$ leftmost of $(\boldsymbol{p}_{i-1}, \boldsymbol{p}_i)$, no angles need to be actually calculated and thus no expensive trigonometric functions need to be evaluated.

In two dimensions this can be implemented by starting with the left-most point $\boldsymbol{p}_0$, and searching for a point $\boldsymbol{p}_1$ such that there is no point left of the line $(\boldsymbol{p}_0, \boldsymbol{p}_1)$, i.e., all

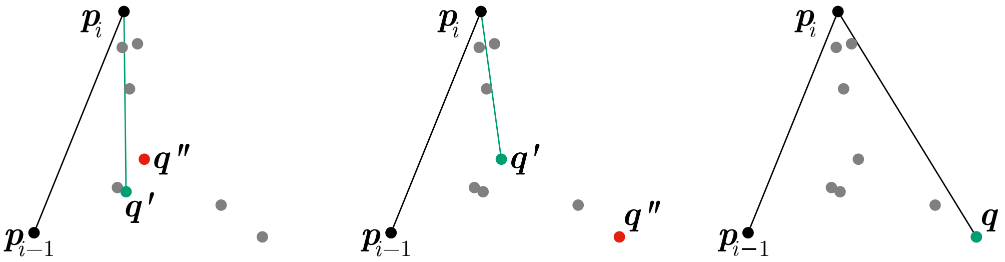


Figure B.2.: Three steps of Jarvis' March algorithm. If a point $\boldsymbol{q}''$ (red) is found left of the line from $\boldsymbol{p}_i$ to the current candidate $\boldsymbol{q}'$ (green), it becomes the new candidate. After testing every point, the one maximizing the angle, $\boldsymbol{q}$, is added to the hull.

triplets of the form $(\boldsymbol{p}_0, \boldsymbol{q}', \boldsymbol{p}_1)$ for all other points $\boldsymbol{q}'$ are counter-clockwise. This $\boldsymbol{p}_1$ is part of the hull. This scheme is repeated with the triple $(\boldsymbol{p}_{i-1}, \boldsymbol{q}, \boldsymbol{p}_i)$, until $\boldsymbol{p}_i = \boldsymbol{p}_0$. At this point the polygon with vertices $\{\boldsymbol{p}_i\}$ is the convex hull.[3] Note that for every of the h points of the hull $n-2$ points need to be checked, resulting in the aforementioned $\mathcal{O}(nh)$ complexity.

B.2.3. Chan's Algorithm

Chan's algorithm [172] is an optimal algorithm for the construction of convex hulls, i.e., an algorithm reaching the theoretical lower bound $\mathcal{O}(n \log h)$ in $d = 2$ and $d = 3$, where n is the number of points and h is the number of points in the resulting hull. Note that ten years prior an algorithm with the same complexity was already presented [173] based on a different, more difficult to implement idea. Here Chan's algorithm is presented, since it is a nice combination of two above introduced algorithms, despite not being used in any study of this thesis. While a version exists for $d = 3$ dimensions, we will limit this description to the $d = 2$ case.

Its basic idea is to split, in a divide-and-conquer spirit, the point set in $k = \lceil n/m \rceil$ subsets each of size less or equal m. The sets and their convex hulls, the *sub-hulls*, may overlap. The sub-hulls are calculated using k times an $\mathcal{O}(m \log m)$ approach like Andrew's monotone chain. Afterwards the sub-hulls are combined with the output dependent Jarvis' march into the global convex hull of all points. The synergy of this split approach comes from the fact that Jarvis' march needs to find the point $\boldsymbol{q}$ maximizing the angle formed with the previous facet of the convex hull $\measuredangle(\boldsymbol{p}_{i-1}, \boldsymbol{p}_i, \boldsymbol{q})$. While in the pure Jarvis' march $\mathcal{O}(n)$ points need to be tested, here, we can get away with less work. The idea, without going into technical details, is that instead of testing every point of each sub-hull we just need to find the point $\boldsymbol{q}$ of each sub-hull such that the line connecting $\boldsymbol{p}_i$ and $\boldsymbol{q}$ is a *tangent* left of the sub-hull. The tangent points of the sub-hulls are the only candidates that need to be tested, since they are left of all other points of the corresponding sub-hull. The tangents are visualized in the example

---

[3] An animation of this process is available at `https://data.schawe.me/jarvis.gif`.

Figure B.3. The trick leading to the reduced work is that finding the tangent point can be done in time $\mathcal{O}(\log m)$ for each of the k sub-hulls. This is achieved with an algorithm similar to binary search.[4] Since the hull of the full problem has h points, this is iterated h times.[5] All in all this results in a time complexity of $\mathcal{O}((n + kh)\log m)$.

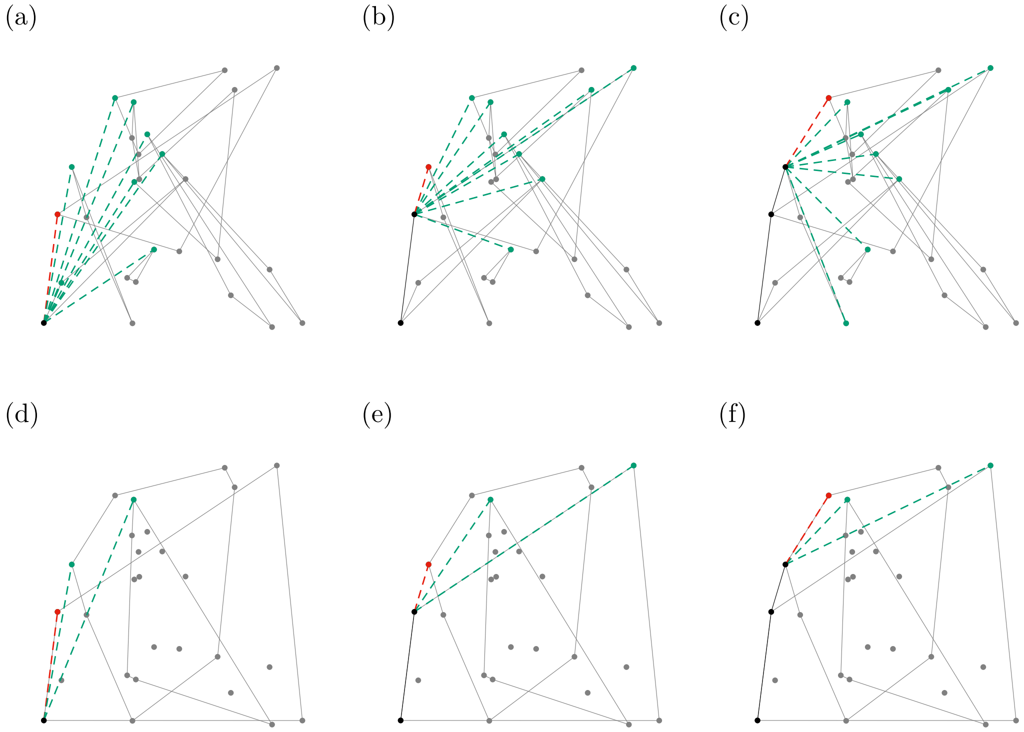


Figure B.3.: Visualization of Chan's algorithm starting with $m = 3$. In (a) to (c) the k candidates and the corresponding tangents are shown in green and the candidate maximizing the angle in red. After $m = 3$ iterations the algorithm is aborted, m is increased to $m = 3^2 = 9$ and the new sub-hulls are calculated. For this iteration again three steps are visualized in (d) to (f).

To achieve the optimal $\mathcal{O}(n \log h)$ run time it is necessary that m is of the order of h. Since h, the number of vertices of the convex hull, is usually not known beforehand, the algorithm is iterated with super-exponentially growing m. In every iteration, only m steps in the Jarvis' phase of the algorithm are performed. Note that if m is growing too slowly, the run time could degrade to $\mathcal{O}\left(n^2\right)$. If the solution is not found, m is squared and the algorithm is started from scratch.

Despite its superior complexity, it is for the relatively small point sets used in this study not necessarily the fastest. A basic, not fine tuned, implementation is consistently slower than the more simple algorithms as shown in Appendix B.2.4 and Figure B.4. Therefore it was not used for any of the studies constituting this thesis.

---

[4] For a more technical look consider References [89, 174].

[5] An animation of this process is available at `https://data.schawe.me/chan.gif`.

B.2.4. Convex Hull Algorithm Implementation for this Study

To decide for an algorithm for our practical study, we should not just select the algorithm with the best time complexity, but have to keep in mind that for this study mainly small point sets of less than $n = 10^4$ are considered, such that the algorithm with the best asymptotic time complexity is not necessarily the fastest for this application. Also we need to preserve the point set for the Markov chain, such that some micro optimizations, like in-place sorting, are not applicable.

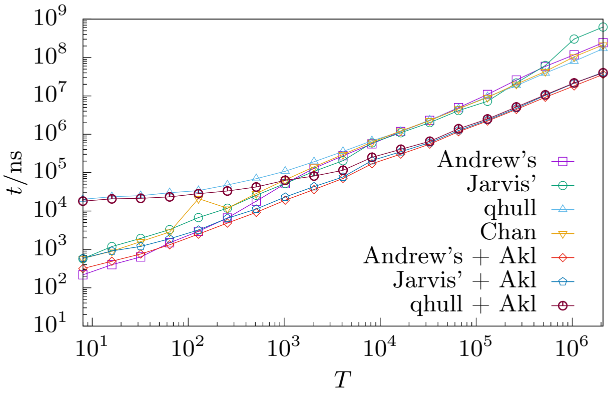


Figure B.4.: Comparison of different algorithms to obtain the convex hull for $d = 2$. Apparently Andrew's Monotone Chain enhanced by Akl's heuristic is the fasted tested implementation, especially in the $n \lesssim 2048$ range where the simulations will be performed. The point set was constructed by a Gaussian random walk.

To decide for an algorithm, comprehensive benchmarks were performed as shown in Figure B.4. For this study, the implementation of Andrew's Monotone Chain enhanced by Akl's heuristic performs best for $d = 2$. For $d = 3$ this algorithm does not work anymore and the Quickhull implementation `qhull` without Akl's heuristic performs best and was used for all $d > 2$. We have no explanation for the anomaly of Chan's algorithm at $T = 128$, the initial value of m is chosen as $m = 100$, a choice of $m = 10$ leads to twice the runtime. The bad runtime of `qhull` for very small instances is probably caused by library call overhead.

Note that technical details, e.g., for handling floating point precision difficulties or degeneracy, are not mentioned in this thesis but played a role in the decision for the well tested `qhull` library instead of an own implementation in higher dimensions.

B.3. Generation of Random Numbers

For every model which is studied as part of this thesis, we need random numbers for its simulation. In fact we only need uniform random numbers and can generate all other distributions from them. Since the generation of good uniform random numbers is a field too wide for this thesis, we will just use a well studied generator without explaining its background. Therefore all random numbers used in the context of this thesis are created using the Mersenne Twister [30]. For a nice introduction into random numbers, Donald E. Knuths "The Art of Computer Programming, Volume 2: Seminumerical Algorithms" [97] is recommended.

Assuming we have a source of good uniform random numbers, we will explore quickly how we can use the uniform random numbers to construct random numbers following different distributions. This will not be an exhaustive review, we will only handle the generation of random numbers distributed according to the distributions we studied in Article A.4.

Exponential For distributions, whose cumulative distribution function can be inverted, it is possible to apply the *inversion method* [175]. So, given a uniform random number $\eta \in [0, 1)$ and the probability density function of the exponential distribution $p(\varepsilon) = \lambda \mathrm{e}^{-\lambda\varepsilon}$ with $\varepsilon \geq 0$, we can generate a random number distributed according to $p(\varepsilon)$ by inverting $\eta = \int_0^\varepsilon \mathrm{d}\varepsilon\, p(\varepsilon) = 1 - \mathrm{e}^{-\lambda\varepsilon}$. This results here in $\varepsilon = -\ln(1-\eta)/\lambda$. Note that the random variables η and $1-\eta$ are indistinguishable, such that this form simplifies to $\varepsilon = -\ln(\eta)/\lambda$.

Gaussian Gaussian random numbers are probably the most used random numbers after uniformly distributed ones. The standard way to generate them is the *Box-Muller method*. It takes two uniform random numbers y_1, y_2 and transforms them into two normal random numbers x_1, x_2 with

$$x_1 = \sqrt{-2\ln(y_1)}\cos(2\pi y_2), \quad x_2 = \sqrt{-2\ln(y_1)}\sin(2\pi y_2). \tag{B.6}$$

The main idea behind the derivation of this method is a mapping to a bivariate Gaussian distribution in polar coordinates, e.g., $x = R\cos(\varphi)$. The probability density in the radial direction is the exponential distribution $q(R) = \mathrm{e}^{-R^2/2}$. Random numbers according to this exponential distribution can be created using the inversion method introduced above, resulting in the first terms of Equation (B.6). The polar angle φ is uniformly distributed $\varphi \in [0, 2\pi)$, which is trivial to obtain using a uniform random number, resulting in the arguments of the trigonometric functions.

The generated Gaussian random numbers follow a standard Gaussian and can be transformed into arbitrary Gaussians through shifts by the desired mean and scaling with the desired standard deviation.

Note that the operations ln, sin, cos and $\sqrt{\cdot}$ are computationally expensive. There is a different formulation of the Box-Muller method avoiding the trigonometric functions [176] or the *Ziggurat method*, which is based on the standard *rejection method* [175] and said to be even more efficient [177]. Though both of which might need many uniformly random numbers to generate one Gaussian random number. Therefore, for our application to construct a Markov chain, it is easier to always store two uniform random numbers per Gaussian ε instead of an undefined number of uniform random numbers, such that we use the Box-Muller method. This can be seen as a speed/convenience trade-off.

Pareto The Pareto distribution is only defined for numbers larger than the parameter $\varepsilon_{\mathrm{min}} > 0$ and its distribution function is $p(\varepsilon) = (\beta - 1)\, \varepsilon_{\mathrm{min}}^{\beta-1} \varepsilon^{-\beta}$, with the parameter β, i.e., it has the form of a power law with the exponent β. This distribution lends itself nicely to the *inversion method*. After integration, we arrive at $\eta = 1 - \left(\varepsilon_{\mathrm{min}}\varepsilon^{-1}\right)^{\beta-1}$ leading to $\varepsilon = \varepsilon_{\mathrm{min}}(1-\eta)^{-1/(\beta-1)}$, to which the same simplification as for the exponential distribution can be applied leading to $\varepsilon = \varepsilon_{\mathrm{min}}\eta^{-1/(\beta-1)}$.

Erlang The distribution $p(\varepsilon) = \varepsilon \mathrm{e}^{-\varepsilon}, \varepsilon \ge 0$ is a special case of the Erlang distribution. The main motivation to examine it is that it shows a different behavior for small values of ε, in particular it behaves like $p(\varepsilon) \sim \varepsilon^{\alpha}$ with the exponent $\alpha = 1$, while for all above distributions $\alpha = 0$. Why this α is important for our application, is explained in Section 4.4.

Again, we can apply the standard inversion method as above to this distribution. Integration leads to

$$\eta = \int_0^{\varepsilon} \mathrm{d}\varepsilon\, \varepsilon \exp -\varepsilon = 1 - (1+\varepsilon)\mathrm{e}^{-\varepsilon}.$$

On the first glance this might look non-invertable, but remembering[6] the definition of the Lambert-W function (cf. Reference [178])

$$z = W\left(z\mathrm{e}^{z}\right),$$

i.e., the Lambert-W function is the inverse of $f(z) = z\mathrm{e}^{z}$, leads to an inversion. Since $f(z) = z\mathrm{e}^{z}$ is not strictly monotone, the inversion is not unique. In this case we need the so called -1-branch W_{-1} of the Lambert-W function. With this we arrive at

$$\varepsilon = -W_{-1}\left(\frac{\eta - 1}{\mathrm{e}}\right).$$

So given a uniform random number η, we can construct one distributed according to the above distribution by evaluation of the -1-branch of the Lambert-W function. For this task most scientific libraries, like the *GNU Scientific Library*, offer a function.

B.4. Biconnected Components

Here we will show the algorithm to find the biconnected components described in Section 6.4 in more detail with an illustrated example. The current node, i.e., the node on top of the stack, which is the only node whose annotations are changed in that step, is marked with a thick border. Edges traversed during the depth-first search, constituting the *depth-first tree* are also marked thick. The full sequence of states leading to the state shown in the main part of this thesis Figure 6.3 is shown in Figure B.5.

---

[6] or being remembered by Mathematica

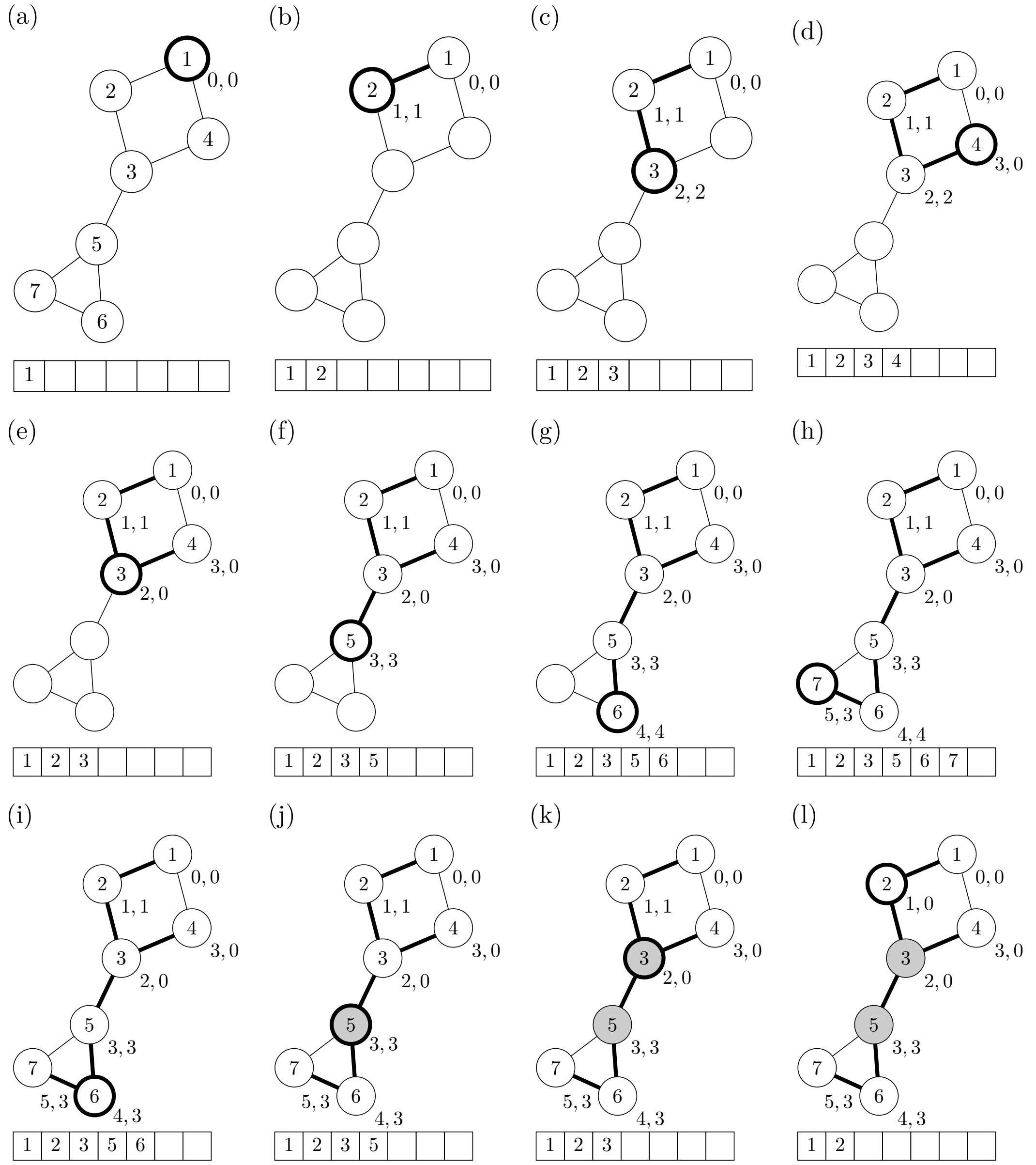


Figure B.5.: Example of Hopcroft and Tarjan's algortihm to find the biconnected components. Each type of step is described in the text. In each panel the current annotated graph is shown and below the current stack of the depth-first search.

Starting at node 1 in Figure B.5(a), the depth-first search proceeds to 2, which is at depth 1, since we have no information gathered yet, we initialize the lowpoint with the depth. The same process is iterated until we do not encounter any unmarked neighbors anymore in Figure B.5(d). At this point, before popping node 4 from the stack, we assign its lowpoint as the minimum of the depths of the neighboring nodes, excluding the node we came from. The same rule leads in Figure B.5(e) to a lowpoint of 0 for node 3, so we can clearly see that the small lowpoint propagates through circles. In Figure B.5(f) node 5 is assigned with depth 3, which is the distance in the depth-first tree to the root node 1. Fast forward to Figure B.5(j), we backtrack to node 5 and notice, before popping it from the stack, that the lowpoint of at least one of its children, i.e., of node 6, is equal to the depth of node 5, such that we mark node 5 as an articulation node. Similarly we notice in Figure B.5(k), before popping node 3 from the stack, that the lowpoint of its child node 5 is greater than the depth of node 3, such that we mark it as an articulation point. This results in the annotated graph of Figure B.5 and the corresponding bicomponents.

In fact, we can use a further auxiliary stack to store edges used for marking the neighbors. When identifying an articulation node and every time we backtrack to a known articulation point, we pop all edges until the edge over which we backtracked to the articulation node. The popped edges define a biconnected component.

C. List of Publications and Preprints and Curriculum Vitae

This is a list of all publications and preprints of which I am a coauthor. Publications and preprints marked by an asterisk $*$ are part of this thesis. The order is not chronological but rather by topic.

- $*$ H. Schawe, A. K. Hartmann, S. N. Majumdar, Convex hulls of random walks in higher dimensions: A large deviation study, *Physical Review E* **96**, 062101 (2017)
- $*$ H. Schawe, A. K. Hartmann, S. N. Majumdar, Large deviations of convex hulls of self-avoiding random walks, *Physical Review E* **97**, 062159 (2018)
- $*$ H. Schawe, A. K. Hartmann, Large Deviations of Convex Hulls of the "True" Self-Avoiding Random Walk, *accepted by Journal of Physics: Conference Series*, arXiv:1808.10698 [cond-mat.stat-mech]
- $*$ J. Börjes, H. Schawe, A. K. Hartmann, Large deviations of the length of the longest increasing subsequence of random permutations and random walks, *Physical Review E* **99**, 042104 (2019)
- $*$ H. Schawe, A. K. Hartmann, S. N. Majumdar, G. Schehr, Ground state energy of noninteracting fermions with a random energy spectrum, *Europhysics Letters* **124**, 40005 (2018)
- $*$ H. Schawe, A. K. Hartmann, Large-deviation properties of the largest biconnected component for random graphs, *The European Physical Journal B* **92**, 73 (2019)
- ◦ H. Schawe, C. Norrenbrock, A. K. Hartmann, Ising Ferromagnets on Proximity Graphs with Varying Disorder of the Node Placement, *Scientific Reports* **7**, 8040 (2017)
- ◦ H. Schawe, R. Bleim, A. K. Hartmann, Phase Transitions of the Typical Algorithmic Complexity of the Random Satisfiability Problem Studied with Linear Programming, *PLOS ONE* **14**, 4 (2019)
- ◦ H. Schawe, A. K. Hartmann, Phase Transitions of Traveling Salesperson Problems solved with Linear Programming and Cutting Planes, *Europhysics Letters* **113**, 30004 (2016)

- H. Schawe, J. K. Jha, A. K. Hartmann, Replica Symmetry and Replica Symmetry Breaking for the Traveling Salesperson Problem, *submitted*, arXiv:1806.08681 [cond-mat.dis-nn]
- M. Jungsbluth, J. Thiele, Y. Winter, H. Schawe, A. K. Hartmann, Vertebrate pollinators: phase transition in a time-dependent generalized traveling-salesperson problem, *submitted*, arXiv:1803.08015 [physics.bio-ph]

Curriculum Vitae

Personal Details

Name	Hendrik Schawe
Address	Immenweg 37 26125 Oldenburg Germany
Email	hendrik.schawe@gmail.com
Web	hendrik.schawe.me
Birthday	July 17 1991, Oldenburg

Research Interests

I am fascinated by simple models exhibiting an unexpected depth. As examples on which I worked, take the easily defined *traveling salesperson problem* turning out to be very hard to solve, the complex behavior of simple models, be it *networks* or *self-avoiding random walks*, or *phase transitions* arising from simple interaction rules like the *Ising model*. Since simulations are my main tool, I am interested in algorithms which enable to study problems, which are on the first glance infeasible, like directly sampling the *large deviation* regime of an observable with *Monte Carlo* methods or finding exact solutions of moderately sized traveling salesperson instances using *linear programming*.

Education

PhD	in the group of Prof. Dr. Alexander K. Hartmann, Carl von Ossietzky Universität Oldenburg, Germany, (Defense: March 19 2019)
M. Sc.	Physics, Carl von Ossietzky Universität Oldenburg, Germany, 2015
B. Sc.	Physics, Carl von Ossietzky Universität Oldenburg, Germany, 2013

Scientific Work

Publications	*Phase Transitions of Traveling Salesperson Problems solved with Linear Programming and Cutting Planes*, H. Schawe, A. K. Hartmann, EPL **113**, 30004 (2016) *Ising Ferromagnets on Proximity Graphs with Varying Disorder of the Node Placement*, H. Schawe, C. Norrenbrock, A. K. Hartmann, Sci. Rep. **7**, 8040 (2017) *Convex hulls of random walks in higher dimensions: A large deviation study*, H. Schawe, A. K. Hartmann, S. N. Majumdar, Phys. Rev. E **96**, 062101 (2017) *Large deviations of convex hulls of self-avoiding random walks*, H. Schawe, A. K. Hartmann, S. N. Majumdar, Phys. Rev. E **97**, 062159 (2018) *Ground-state energy of noninteracting fermions with a random energy spectrum*, H. Schawe, A. K. Hartmann, S. N. Majumdar, Grégory Schehr, EPL **124**, 40005 (2018) *Large deviations of the length of the longest increasing subsequence of random permutations and random walks*, J. Börjes, H. Schawe, A. K. Hartmann, Phys. Rev. E **99**, 042104 (2019) *Large-deviation properties of the largest biconnected component for random graphs*, H. Schawe, A. K. Hartmann, EPJB **92**, 73 (2019) *Phase Transitions of the Typical Algorithmic Complexity of the Random Satisfiability Problem Studied with Linear Programming*, H. Schawe, R. Bleim, A. K. Hartmann, PLOS ONE **14**, 4 (2019)

Accepted	*Large Deviations of Convex Hulls of the "True" Self-Avoiding Random Walk*, H. Schawe, A. K. Hartmann, Journal of Physics: Conference Series arXiv:1808.10698
Under Revision	*Vertebrate pollinators: phase transition in a time-dependent generalized traveling-salesperson problem*, M. Jungsbluth, J. Thiele, Y. Winter, H. Schawe, A. K. Hartmann, arXiv:1803.08015 *Replica Symmetry and Replica Symmetry Breaking for the Traveling Salesperson Problem*, H. Schawe, J. K. Jha, A. K. Hartmann, arXiv:1806.08681 *On the asymptotic behavior of the length of the longest increasing subsequences of random walks*, J. Ricardo G. Mendonça, H. Schawe, A. K. Hartmann, arXiv:1907.00486

Skills

Languages	German, English, Latin C++, Python, C, Rust

Academic Events

Conferences	DPG-Frühjahrstagung, Regensburg, Germany, 2019 IUPAP Conference on Computational Physics, Davis, CA, USA, 2018 DPG-Frühjahrstagung, Berlin, Germany, 2018 CompPhys17, 18th International NTZ-Workshop on New Developments in Computational Physics, Leipzig, Germany, 2017 DPG-Frühjahrstagung, Dresden, Germany, 2017 CompPhys16, 17th International NTZ-Workshop on New Developments in Computational Physics, Leipzig, Germany, 2016 DPG-Frühjahrstagung, Regensburg, Germany, 2016 DPG-Frühjahrstagung, Dresden, Germany, 2014
Research Visits	Laboratoire de Physique Théorique et Modèles Statistiques (LPTMS), Orsay, France, 2016
Summer Schools	Fundamental Problems in Statistical Physics XIV (FPSP XIV), Bruneck, Italy, 2017 Bad Honnef Physics School *Computational Physics of Complex and Disordered Systems*, Bad Honnef, Germany, 2015, 5th International Summer School on Modern Compuational Science *Compuational Quantum Chemistry*, Oldenburg, Germany, 2014

July 9, 2019

Hendrik Schawe

Acknowledgments

This part of my life would not have been possible without other people. Therefore, I want to thank Alexander K. Hartmann for the opportunities and the freedom to research a diverse collection of problems, to extend my skills and social network on summer schools and during a visit to another research institute, and to present my results at conferences.

I want to thank Christoph Norrenbrock, Pascal Fieth, Marcel Kahlen and Hendrik Neumann, for reading parts of this dissertation and supplying feedback. Also I want to thank Hauke Fajen, Jörn Börjes, Sebastian von Ohr, Roman Bleim, Charlotte Beelen, Daniel Grujic, Timo Dewenter, Wiebke Staffelt, Yannick Feld and Christoph Polle, who were the members of the working group during this time, for the pleasant atmosphere.

Also, I want to thank Satya N. Majumdar for the collaboration on multiple projects, as well as Grégory Schehr, Jithesh Jha, Roman Bleim and Christoph Norrenbrock with whom I wrote, submitted and published research papers.

I want to thank the DFG for funding, Stefan Krautwald for keeping my workstation operable and Stefan Harfst and the GWDG for ensuring the availability of the respective compute clusters they manage.

Finally, I want to thank my friends and family keeping up the balance of my

Danksagung

An dieser Stelle möchte ich all jenen danken, die zu diesem Teil meines Lebens beigetragen haben. Alexander K. Hartmann, dafür dass er mir die Möglichkeiten und Freiheiten gegeben hat eine Ansammlung abwechslungsreicher Probleme zu erforschen, auf Sommerschulen und einem Forschungsbesuch an einem anderen Institut meine Fähigkeiten und mein soziales Netzwerk auszubauen und auf vielen Tagungen und Konferenzen meine Ergebnisse vorzustellen.

Christoph Norrenbrock, Pascal Fieth, Marcel Kahlen und Hendrik Neumann, für das Probelesen der meisten Teile dieser Dissertation und zusammen mit Hauke Fajen, Jörn Börjes, Sebastian von Ohr, Roman Bleim, Charlotte Beelen, Daniel Grujic, Timo Dewenter, Wiebke Staffelt, Yannick Feld und Christoph Polle, die während meiner Promotion Mitglieder der Arbeitsgruppe waren, dafür dass ich gerne Zeit in der Universität verbracht habe.

Weiterhin gilt mein Dank Satya N. Majumdar für die Zusammenarbeit an mehreren Projekten dieser Arbeit, sowie Grégory Schehr, Jithesh Jha, Roman Bleim und Christoph Norrenbrock mit denen zusammen ich Manuskripte geschrieben, eingereicht und veröffentlicht habe.

Ich bedanke mich bei der DFG für die Finanzierung meiner Forschung, Stefan Krautwald dafür, dass mein Arbeitscomlife. Without them this thesis would not exist.

puter immer funktionsfähig war und Stefan Harfst sowie der GWDG dafür, dass sie jeweils die Supercomuter unter ihrer Verwaltung für meine Simulationeen funktionstüchtig gehalten haben.

Außerdem möchte ich mich bei meinen Freunden und meiner Familie bedanken – ohne sie existierte diese Dissertation nicht.

Hiermit versichere ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.

(Hendrik Schawe)